

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Vitjan Zavrtanik

Učinkovito generiranje hipotetičnih slikovnih regij za detekcijo polipov

DIPLOMSKO DELO

UNIVERZITETNI ŠTUDIJSKI PROGRAM
PRVE STOPNJE
RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: doc. dr. Matej Kristan
SOMENTOR: izr. prof. dr. Jasna Maver

Ljubljana, 2016

To delo je ponujeno pod licenco *Creative Commons Priznanje avtorstva-Deljenje pod enakimi pogoji 2.5 Slovenija* (ali novejšo različico). To pomeni, da se tako besedilo, slike, grafi in druge sestavine dela kot tudi rezultati diplomskega dela lahko prosto distribuirajo, reproducirajo, uporabljajo, priobčujejo javnosti in predelujejo, pod pogojem, da se jasno in vidno navede avtorja in naslov tega dela in da se v primeru spremembe, preoblikovanja ali uporabe tega dela v svojem delu, lahko distribuira predelava le pod licenco, ki je enaka tej. Podrobnosti licence so dostopne na spletni strani creativecommons.si ali na Inštitutu za intelektualno lastnino, Streliška 1, 1000 Ljubljana.



Izvorna koda diplomskega dela, njeni rezultati in v ta namen razvita programska oprema je ponujena pod licenco GNU General Public License, različica 3 (ali novejša). To pomeni, da se lahko prosto distribuira in/ali predeluje pod njenimi pogoji. Podrobnosti licence so dostopne na spletni strani <http://www.gnu.org/licenses/>.

Besedilo je oblikovano z urejevalnikom besedil \LaTeX .

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Številne znanstvene analize temeljijo na ročni anotaciji podatkov, kar zahteva veliko napora in zbranosti ter je podvrženo človeškim napakam. To je še posebej očitno pri analizah populacije meduz, kjer znanstveniki z ročnim označevanjem stotin polipov na podvodnih fotografijah ocenjujejo njeno velikost. Postopke štetja polipov je moč avtomatizirati s postopki računalniškega vida, ki temeljijo na dvostopenjski detekciji. Prva stopnja obsega generiranje hipotetičnih regij polipov, druga stopnja pa verifikacijo hipotez z močnim klasifikatorjem. V diplomski nalogi se osredotočite na prvo stopnjo. Predlagajte za polipe specializirane detektorje regij in jih primerjajte z obstoječimi splošnimi postopki za detekcijo. Detektorje analizirajte kvalitativno in kvantitativno.

Zahvaljujem se mentorju doc. dr. Mateju Kristanu za potrpežljivost, izjemno odzivnost ter odlične napotke in ideje pri izdelavi diplomskega dela. Prav tako se za strokovno pomoč zahvaljujem tudi somentorici izr. prof. dr. Jasni Maver. Zahvaljujem se tudi družini in puncu Anji za izjemno podporo, pomoč in vzpodbudo.

Kazalo

Povzetek

Abstract

1	Uvod	1
1.1	Motivacija	1
1.2	Cilji diplomske naloge	2
1.3	Sorodna dela	3
1.4	Prispevki	5
1.5	Struktura dela	5
2	Metode	7
2.1	Usmerljivi filtri	7
2.2	Radialnost	15
2.3	Segmentacija Felzenszwalb	17
2.4	Selective Search	20
2.5	Algoritem ACF	22
3	Predlagana metoda	27
3.1	Generiranje maske	28
3.2	Predlaganje regij	30
3.3	Kombinacija z algoritmom Felzenszwalb	32
4	Eksperimentalna analiza	33
4.1	Podatkovna zbirka	33

4.2	Ocenjevanje predlaganih regij	37
4.3	Izbira parametrov RRD	42
4.4	Primerjava rezultatov	49
4.5	Rezultati metode ACF	52
4.6	Rezultati Selective Search	53
4.7	Analiza metode RRD	54
4.8	Rezultati kombinacije metod RRD in Felzenszwalb	58
4.9	Kombinacija metod ACF in RRD	59
4.10	Prednosti in slabosti metode RRD	60
5	Sklep	63
5.1	Možne izboljšave	64
	Literatura	66

Seznam uporabljenih kratic

kratica	angleško	slovensko
IOU	intersection over union	presek čez unijo
RRD	Radial Region Detection	Detekcija radialnih regij
AR	Average Recall	Povprečni priklic
DoG	Difference of Gaussians	Razlika Gaussovih filtrov
MST	Minimum Spanning Tree	minimalno vpeto drevo
AR3	Average Recall between IOU 0.3 and 1	Povprečni priklic med IOU 0.3 in 1
ACF	Aggregated Channel Features	agregirane značilke po kanalih
HOG	Histogram of oriented gradients	histogram orientiranih gradientov

Povzetek

Naslov: Učinkovito generiranje hipotetičnih slikovnih regij za detekcijo polipov

Avtor: Vitjan Zavrtanik

V tej nalogi naslavljamo problem detekcije polipov v slikah. Moderne metode detekcije so pogosto sestavljene iz dveh sklopov. Najprej se na obravnavani sliki hitro generirajo hipotetične regije na mestih, kjer naj bi se nahajali iskani objekti, nato pa se na predlaganih lokacij izvede še verifikacija hipotez z močnim klasifikatorjem. V tem delu se osredotočamo na prvi korak detekcij in poskušamo optimizirati novo metodo za reševanje problema predlaganja regij za detekcijo polipov na slikah ostrig. Rezultate predlagane metode smo primerjali z drugimi metodami za predlaganje regij kot sta ACF in Selective Search. Predlagali smo modificirano mero za ocenjevanje uspešnosti, ki bazira na meri AR. Problem detekcije polipov na slikah je težko rešljiv zaradi veliko različnih faktorjev, ki vplivajo na izgled polipa, ki ga je potrebno prepoznati. Polipi so različnih velikosti, so različno orientirani, se lahko zaradi svoje prosojnosti zlijejo z ozadjem in se velikokrat tudi prekrivajo kar zanesljivo detekcijo posameznih polipov lahko onemogoči.

Ključne besede: predlaganje regij, polipi, detekcija objektov.

Abstract

Title: Efficient region proposals for polyp detection

Author: Vitjan Zavrtanik

In this thesis we deal with the issue of polyp detection in images of oysters. Modern methods of object detection are often composed of two parts. Firstly we use a fast region proposal method to generate hypothetical regions in places where objects are located with a higher probability. We then use a strong classifier to verify the proposed hypothetical regions. We address the issue of the first part of the object detection pipeline and we propose a new region proposal method for the purpose of detecting polyps in images of oysters. We compared the results of our method with other region proposal methods such as ACF and Selective Search. We also propose a metric for region proposal method performance based on the AR metric. Polyp detection is a hard problem to solve due to the visual properties of individual polyps as they often vary in size, orientation and opacity which causes them to blend with their surroundings and can often make reliable detection difficult.

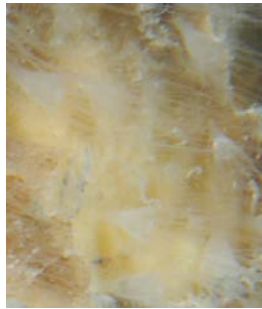
Keywords: region proposal, polyps, object detection.

Poglavje 1

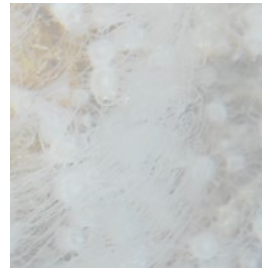
Uvod

1.1 Motivacija

Meduze del življenjskega cikla preživijo kot polipi pritrjeni na morsko dno, velikokrat pa se nahajajo na ostrigah [1]. Biologi, ki se z njimi ukvarjajo uporabljajo podatke o številu polipov za napovedovanje populacije meduz [1]. Za pridobivanje podatkov o številu polipov je potrebno štetje polipov na večji količini slik ostrig, kar se izkaže za dolgotrajen proces, ki je zaradi nekaterih značilnosti polipov za ljudi težaven. Polipi so lahko različnih oblik, velikosti, so na slikah različno orientirani, se velikokrat prekrivajo kot vidimo na Sliki 1.1b in se na slikah zaradi svoje prosojnosti včasih zlijejo z okoljem kot vidimo na Sliki 1.1a. Dodaten problem predstavlja število polipov na slikah saj jih je v povprečju na sliki več kot tisoč. Računalniški vid ima v biologiji veliko drugih možnosti uporabe pri avtomatizaciji nekaterih procesov, ki bi se v nasprotnem primeru izvajali ročno [35].



(a)



(b)

Slika 1.1: a) Slaba vidljivost polipov zaradi prosojnosti polipov ter svetlega ozadja. b) Polipe se težko loči med sabo zaradi delnega medsebojnega prekrivanja.

1.2 Cilji diplomske naloge

Za detekcijo objektov na slikah se je dalj časa uporabljalo metode drsečih oken, kjer se je na vsaki možni lokaciji v sliki apliciralo klasifikator v različnih velikostih in tako ugotavljalo ali nek določen objekt na sliki obstaja [15, 34, 47]. Glavni problem uporabe drsečih oken je, da je za večjo zanesljivost potrebno preveriti zelo veliko različnih lokacij na sliki, kar pa se izkaže za časovno zahtevno. V primeru detekcije objektov, ki se na sliki pojavljajo v različnih velikostih se število potrebnih preverjenih oken še drastično zviša. Z razvojem močnejših in hitrejših detekcijskih mrež se je predlaganje regij izkazalo kot nujnost za razvoj detekcije objektov saj je bilo pokazano, da je velika količina lokacij, ki jih je potrebno preveriti, ozko grlo tega področja [37, 19]. Tako se kot predprocesni korak za detekcijo velikokrat najprej uporabi metode predlaganja regij, ki v veliko krajšem času generirajo predloge lokacij kjer se nahajajo objekti velikokrat brez upoštevanja razreda objekta [21]. To zmanjša število možnih lokacij v sliki, ki jih je potrebno preveriti, kar bistveno pospeši detekcijo oziroma omogoči uporabo veliko zahtevnejših klasifikatorjev.

Pričujoče delo se ukvarja s predlaganjem regij polipov na slikah ostrig, kot korak predprocesiranja, kar omogoča uporabo zahtevnejših klasifikatorjev pri

razvoju sistema za avtomatsko štetje polipov. Cilj je predlagati učinkovito metodo za generiranje hipotetičnih regij za boljšo detekcijo polipov pri čimer je bilo več metod prilagojenih za delovanje na danem problemu. Uspešnost posameznih metod je bila nato ocenjena, metode pa tako primerjane med seboj.

1.3 Sorodna dela

Zaradi vrste možnih aplikacij detekcije objektov in posledične velike potrebe po njegovem pospeševanju je predlaganje regij (ang. region proposal) popularno raziskovalno področje [14, 49, 44, 9, 30, 38]. Veliko del se ne ukvarja z detekcijo polipov meduz vendar pa lahko zaradi vizualne podobnosti polipov s celicami med sorodna dela štejemo tudi raziskovalna dela, ki se ukvarjajo z detekcijo in štetjem celic na slikah mikroskopiranega tkiva. Slike polipov na morskemu dnu in slike celic so si podobne glede na spreminjajoče se ozadja in majhne variacije v barvah celic oziroma polipov. Poleg tega so lahko polipi in celice različnih oblik in velikosti, prav tako pa se lahko na slikah prekrivajo. Detekcija celic je zaradi zelo velikega števila nerešenih problemov v biologiji in medicini že več desetletij aktivno področje v raziskavah, kar je spodbudilo razvoj velikega števila algoritmov [7, 4, 3, 20], ki uporabljajo vrsto različnih pristopov katerih uspešnost je spremenljiva s tipom, gostoto, velikostjo in obliko celic ter z vrsto mikroskopije, ki je bila za slikanje uporabljena [33]. Nekateri izmed najbolj popularnih pristopov za segmentacijo slik celic so binarizacija glede na intenzivnost v sivinski sliki, detekcija značilnk [3], morfološko filtriranje [20], akumulacija regij [20] in prileganje spremenljivih modelov [32]. Veliko metod uporablja kombinacijo naštetih pristopov [2, 33].

Večino izmed trenutno najbolj popularnih in uspešnih metod predlaganja regij lahko razdelimo glede na dva pristopa h generiranju predlogov. Metode predlaganja regij predlagajo dele slike na katerih se z višjo verjetnostjo nahajajo objekti. Potencialne lokacije objektov so velikokrat označene z očrtanim

pravokotnikom, ki označuje pozicijo in velikost objekta (ang. bounding box). Prvi pristop k predlaganju regij je ocenjevanje posameznih oken pri čemer se iz slike vzorči večje število oken na katerih se nato oceni s kakšno verjetnostjo se v njih nahaja objekt. Primera tega pristopa sta EdgeBoxes [14] in Bing [9]. Obstaja še pristop grupiranja v katerem metode sprva generirajo začetne segmente slike, ki jih nato združujejo v večje segmente, ki bolj verjetno zajemajo nek objekt glede na različne kriterije. Primer grupiranja je SelectiveSearch [45, 46].

Ob porastu uporabe predlaganja regij se je seveda pojavila tudi potreba po analizi uspešnosti pristopov vendar pa ustaljene metode za merjenje uspešnosti posameznih algoritmov na podatkovnih zbirkah ni. V delu [21] se predlaga AR (ang. Average Recall) kot standardno metodo za ocenjevanje dobrih detekcijskih predlogov kjer detekcije kot dobre upoštevajo le, če se z ročnimi anotacijami objektov dovolj dobro prekrivajo. Za metriko prekrivanja so uporabili razmerje IOU (ang. Intersection Over Union), predlagano regijo pa upoštevajo le v primeru, da razmerje med presekom in unijo regije ter ročne anotacije presega 0.5. AR je tako definirana kot površina pod krivuljo na grafu priklica v odvisnosti od meje IOU, kjer je meja IOU večja ali enaka 0.5. V delih tudi pokažejo močno korelacijo med AR in uspešnostjo detekcije objektov na lokacijah pridobljenih iz pripadajoče metode predlaganja regij [21, 22].

S problemom detekcije polipov se ukvarja tudi diplomsko delo [41]. Za predlaganje regij se v delu uporablja algoritem ACF (aggregated channel features) [11], ki je bil naučen na slikah polipov. Za pozitivne primere delo šteje vse generirane regije katerih IOU dosega vrednost 0.2. S problemom štetja polipov se ukvarja še diplomsko delo [6], ki z uporabo večnivojskega perceptrona klasificira posamezna okna kot ozadje ali kot polip in nato vsa okna, klasificirana kot ozadje, odstrani. Sliko z odstranjenim ozadjem nato binarizira [8] ter z metodo drsečih oken kot polip klasificira vsako okno, ki

vsebuje manj kot 5% praznega prostora. Regijo klasificirano kot polip nato odstrani iz binarne slike. Število polipov pridobi z modelom napovedovanja števila polipov, ki poskuša napovedati napako klasifikatorja.

1.4 Prispevki

Prispevki tega diplomskega dela so analiza večih pristopov predlaganja regij ter ocenjevanje posameznih metod pri problemu detekcije in štetja polipov. Največji prispevek diplomske naloge uvedba nove metode za detekcijo polipov RRD (ang. radial region detection), njeno apliciranje na dani problem ter ocenjevanje uspešnosti. Izvedenih je bilo večje število eksperimentov, ki so preverjali več parametrov RRD z namenom ugotavljanja njihovih optimalnih vrednosti za reševanje problema predlaganja regij polipov. Metodo RRD smo prilagodili za dobro delovanje na dani podatkovni bazi slik polipov na ostrigah. Poleg RRD smo na bazi slik ostrig preizkusili tudi druge algoritme, jih prilagodili za reševanje problema detekcije polipov ter njihove rezultate analizirali. Uvedli smo tudi novo mero za ocenjevanje uspešnosti metod za reševanje problema predlaganja regij, ki bazira na meri AR [21].

1.5 Struktura dela

V Poglavlju 2 so opisane uporabljene metode. Opisano je delovanje primerjanih metod in morebitne metode računalniškega vida, ki jih pri svojem delovanju uporabljajo. Metoda RRD je podrobneje opisana v Poglavlju 3. V Poglavlju 4 je opisano izvedeno eksperimentalno delo in rezultati tega. Poglavlje 5 pa le povzame rezultate raziskovalnega dela ter predlaga možnosti za nadaljnjo delo.

Poglavje 2

Metode

To poglavje opisuje delovanje različnih tehnik računalniškega vida, ki jih naš pristop uporablja. Opisuje tudi samostojne metode predlaganja regij, ki jih uporabljamo v kombinaciji s predlagano metodo ali pa jih uporabljamo za primerjavo. Poglavje 2.1 opisuje delovanje in teorijo usmerljivih filtrov [17]. Poglavje 2.2 se osredotoča na radialnost [31]. Algoritem Felzenszwalb [16] je opisan v Poglavju 2.3, algoritem SelectiveSearch [44] pa v Poglavju 2.4. Poglavje 2.5 opisuje delovanje algoritma ACF [11].

2.1 Usmerljivi filtri

Usmerljivi filtri [17] se lahko uporabljajo pri opravljanju različnih nalog računalniškega vida, ki imajo v svoje reševanje vključene orientirane filtre kot so detekcija robov, analiza tekstur ali analiza gibanja [23, 17, 42]. V primeru, da želimo pridobiti odziv filtra pri mnogih orientacijah, imamo le nekaj možnosti. Na sliko lahko apliciramo več verzij istega filtra, ki se med seboj razlikujejo po rotaciji, vendar pa je to neučinkovito, saj moramo postopek konvolucije velikokrat ponoviti. Bolj učinkovit pristop je uporaba usmerljivih filtrov [17] katerih glavna lastnost je, da lahko z le manjšo količino filtrov z različnimi orientacijami ter s pravimi interpolacijskimi funkcijami

izračunamo odziv filtra pri poljubni orientaciji. Orientiran filter pod poljubnim kotom je torej izražen kot linearna kombinacija baznih filtrov, pravila za interpolacijo pa določajo interpolacijske funkcije. Za to, da je nek filter usmerljiv mora držati enačba:

$$f^\theta(x, y) = \sum_{j=1}^M k_j(\theta) f^{\theta_j}(x, y), \quad (2.1)$$

kjer je $f^\theta(x, y)$ funkcija obrnjena za kot θ . Izražena je z linearno kombinacijo drugače orientiranih verzij same sebe $f^{\theta_j}(x, y)$ z interpolacijskimi funkcijami $k_j(\theta)$. Orientacijo baznih filtrov $f^{\theta_j}(x, y)$ predstavlja θ_j . V enačbi je M minimalna količina baznih filtrov potrebnih za usmerljivost $f(x, y)$. M se za posamezno funkcijo lahko določi s številom neničelnih koeficientov $a_n(r)$ pri razširitvi funkcije v Furierovo zaporedje v polarnem kotu ϕ :

$$f(r, \phi) = \sum_{n=-N}^N a_n(r) e^{in\phi}. \quad (2.2)$$

Za to, da je filter usmerljiv moramo torej vedeti kateri so bazni filtri $f^{\theta_j}(x, y)$, koliko jih je (M) ter katere so interpolacijske funkcije $k_j(\theta)$, da je enačba (2.2) zadovoljena.

V našem primeru na isti sliki izvedemo več usmerljivih filtrov ter nato sestavimo masko glede na maksimume določenih filtrov. Za primer usmerljivega filtra lahko predstavimo prvi odvod dvodimenzionalne Gaussove funkcije, ki je hkrati tudi eden izmed filtrov uporabljenih v predlagani metodi. Dvodimenzionalni Gaussov filter lahko zapišemo kot:

$$G(x, y) = e^{-\left(\frac{x^2}{2\sigma^2} + \frac{y^2}{2\sigma^2}\right)}, \quad (2.3)$$

kjer je $G(x, y)$ vrednost dvodimenzionalne Gaussove funkcije v točko (x, y) . Standardni odklon Gaussove funkcije je predstavljen s σ . Bazna filtra potrebna za usmerjanje odvoda Gaussove funkcije sta njegova odvoda po x in y oziroma odvod Gaussove funkcije usmerjen v 0° in 90° okoli središča:

$$G'_{0^\circ}(x, y) = \frac{\delta}{\delta x} e^{-(\frac{x^2}{2\sigma^2} + \frac{y^2}{2\sigma^2})} = -\frac{x}{\sigma^2} e^{-(\frac{x^2}{2\sigma^2} + \frac{y^2}{2\sigma^2})}, \quad (2.4)$$

$$G'_{90^\circ}(x, y) = \frac{\delta}{\delta y} e^{-(\frac{x^2}{2\sigma^2} + \frac{y^2}{2\sigma^2})} = -\frac{y}{\sigma^2} e^{-(\frac{x^2}{2\sigma^2} + \frac{y^2}{2\sigma^2})}. \quad (2.5)$$

Za interpolacijske funkcije pri poljubnem kotu θ uporabimo:

$$k1 = \cos(\theta), k2 = \sin(\theta). \quad (2.6)$$

Filter odvoda Gaussove funkcije orientiranega v poljubno smer pa lahko izrazimo z linearno kombinacijo baznih filtrov G'_{0° in G'_{90° :

$$G'_\theta = G'_{0^\circ} \cos(\theta) + G'_{90^\circ} \sin(\theta). \quad (2.7)$$

Ker je konvolucija linearna operacija, jo lahko na sliki izvedemo s filtrom s poljubno orientacijo tako, da najprej izvedemo konvolucijo z baznimi filtri G'_{0° in G'_{90° , nato pa uporabimo $k1$ in $k2$ za interpolacijo pri linearni kombinaciji rezultatov konvolucije, ki je enaka linearni kombinaciji za sintetiziranje filtra G'_θ pri poljubni orientaciji. Če je $*$ operator konvolucije, bi lahko konvolucijo z usmerjanjem odvoda Gaussovega filtra na sliki I zapisali kot:

$$C_{0^\circ} = G'_{0^\circ} * I, \quad (2.8)$$

$$C_{90^\circ} = G'_{90^\circ} * I, \quad (2.9)$$

$$C_\theta = C_{0^\circ} \cos(\theta) + C_{90^\circ} \sin(\theta), \quad (2.10)$$

kjer je θ kot, ki ga lahko izračunamo iz orientacije gradienta slike. Gradient slike ima največji odziv tam, kjer se intenzivnost najhitreje spreminja [24]. Gradient slike I je podan z enačbo:

$$\Delta I = \begin{bmatrix} \frac{\delta I}{\delta x} \\ \frac{\delta I}{\delta y} \end{bmatrix} = \begin{bmatrix} C_{0^\circ} \\ C_{90^\circ} \end{bmatrix}. \quad (2.11)$$

Orientacijo gradienta slike lahko tako izračunamo :

$$\theta = \tan^{-1}\left(\frac{\frac{\delta I}{\delta y}}{\frac{\delta I}{\delta x}}\right) = \tan^{-1}\left(\frac{C_{90^\circ}}{C_{0^\circ}}\right). \quad (2.12)$$

Rezultat usmerjenega filtriranja z odvodom Gaussove funkcije vidimo na Sliki 2.1.

V predlagani metodi se poleg usmerljivega filtra prvega odvoda Gaussove funkcije uporablja še druge filtre:

F.1 filter DoG [29],

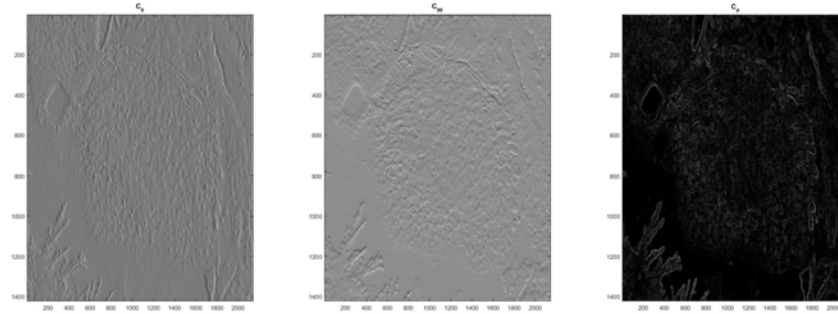
F.2 Usmerljiv filter prvega odvoda Gaussove funkcije,

F.3 Usmerljiv filter drugega odvoda Gaussove funkcije po x ,

F.4 Usmerljiv filter drugega odvoda Gaussove funkcije po y ,

F.5 Usmerljiv filter drugega odvoda Gaussove funkcije po x in y .

Postopek za usmerjanje drugega odvoda Gaussovega filtra je zelo podoben primeru usmerjanja prvega odvoda dvodimenzionalne Gaussove funkcije



Slika 2.1: Od leve proti desni. Rezultat konvolucije s prvim odvodom Gaussove funkcije po x C_{0° , rezultat konvolucije s prvim odvodom Gaussove funkcije po y C_{90° , linearna kombinacija C_{0° in C_{90° .

le, da se izvede za drugačen filtre. V metodi se uporabljajo drugi odvodi Gaussovega filtra, ki jih pridobimo s tem, da Gaussovo funkcijo odvajamo dvakrat. Za **F.3** moramo dvakrat odvajati po X, za **F.4** moramo dvakrat odvajati po Y, za **F.5** pa moramo najprej odvajati po X nato pa še enkrat po Y. Za vse uporabljene filtre, ki so v osnovi drugi odvodi Gaussove funkcije velja, da imajo enake interpolacijske funkcije ter, da se orientacija θ pri vseh izračuna po enaki formuli.

Za usmerjanje **F.3** se kot bazne filtre uporablja več različnih rotacij drugega odvoda Gaussove funkcije. Z $G_{xx}^{0^\circ}$ označimo dvojni odvod Gaussovega filtra po X obrnjenega za 0° . Kot bazne filtre pri **F.3** uporabljamo $G_{xx}^{0^\circ}$, $G_{xx}^{60^\circ}$ in $G_{xx}^{120^\circ}$ saj se v praksi, zaradi simetričnosti ter robustnosti proti šumu uporablja enakomerno rotirane bazne filtre med 0 in π [17]:

$$G_{xx}^{0^\circ} = \frac{-1}{\sigma^2} + \frac{x^2}{\sigma^4} e^{-\left(\frac{x^2}{2\sigma^2} + \frac{y^2}{2\sigma^2}\right)}, \quad (2.13)$$

kjer je $G_{xx}^{0^\circ}$ drugi odvod Gaussove funkcije, ki ga pridobimo z dvakratnim odvajanjem po x. Za pridobitev ostalih baznih filtrov se $G_{xx}^{0^\circ}$ še dvakrat zarotiramo do $G_{xx}^{60^\circ}$ in $G_{xx}^{120^\circ}$ nato pa se kot v prej podanem primeru za vsak bazni filter izvede konvolucijo na sliki ter uporabi interpolacijska pravila za

sintezo usmerjenega filtriranja. Interpolacijsko pravilo je v našem primeru:

$$k_j(\theta) = \frac{1}{3}(1 + \cos(2(\theta - \theta_j))), \quad (2.14)$$

kjer je posamezna interpolacijska funkcija $k_j(\theta)$ izračunana za vsak kot θ_j , ki predstavlja rotacije baznih filtrov torej za 0° , 60° in 120° oziroma v radianih v tem vrstnem redu 0 , $\frac{2\pi}{3}$ in $\frac{4\pi}{3}$ radianov. Orientacija θ pa se v tem primeru izračuna z enačbo:

$$\theta = \tan^{-1}\left(\frac{-(G_{xx}^{120^\circ} - G_{xx}^{60^\circ}) * \sqrt{3}}{-(G_{xx}^{120^\circ} + G_{xx}^{60^\circ} - 2G_{xx}^{0^\circ})}\right). \quad (2.15)$$

Rezultat konvolucije slike s filtri $G_{xx}^{0^\circ}$, $G_{xx}^{60^\circ}$ in $G_{xx}^{120^\circ}$ označimo s $C_{xx}^{0^\circ}$, $C_{xx}^{60^\circ}$ in $C_{xx}^{120^\circ}$. Konvolucijo z usmerjenim filtrom G_{xx}^θ lahko sintetiziramo z linearno kombinacijo interpolacijskega pravila (2.14) ter baznih filtrov:

$$C(\theta) = \frac{1}{3}([1 + \cos(2\theta)]G_{xx}^{0^\circ} + [1 + \cos(2(\theta - \frac{2\pi}{3}))]G_{xx}^{60^\circ} + [1 + \cos(2(\theta - \frac{4\pi}{3}))]G_{xx}^{120^\circ}). \quad (2.16)$$

Tudi za **F.5** velja enak postopek kot za **F.3**. Pri obeh filterih se orientacija θ izračuna po enačbi (2.15), uporabljata pa tudi enaka interpolacijska pravila (2.14). Razlikujeta pa se seveda v baznih filterih saj **F.5** najprej odvaja po x nato pa po y . Tudi **F.5** svoje bazne filtre pridobi z rotacijo za 0° , 60° in 120° :

$$G_{xy}^{0^\circ} = \frac{y}{\sigma^2} \frac{x}{\sigma^2} e^{-(\frac{x^2}{2\sigma^2} + \frac{y^2}{2\sigma^2})}, \quad (2.17)$$

kjer je $G_{xy}^{0^\circ}$ drugi odvod Gaussove funkcije, ki ga pridobimo tako, da (2.3) najprej odvajamo po x nato pa po y . Za detekcijo interesnih točk se pogosto uporablja DoG (difference of Gaussians) [27, 26], ki deluje tako, da se na sliki izvede konvolucijo z dvema Gaussovimi filtri, ki se med seboj razlikujeta v parametru σ nato pa se rezultata konvolucije med seboj odšteje.

DoG ima visok odziv na regijah z večjimi spremembami v intenziteti zaradi česar se med drugim uporablja tudi za detekcijo robov [5]. Ker z uporabo pravih parametrov DoG dobro predstavi svetle regije, ki predstavljajo polipe, ga vključimo v kombinacijo filtrov metode:

$$C_{DoG} = I * G_{\sigma} - I * G_{\sigma K}, \quad (2.18)$$

kjer je C_{DoG} rezultat konvolucije na sliki I z razliko filtrov G_{σ} in $G_{\sigma K}$. Filter G_{σ} je enak:

$$G_{\sigma}(x, y) = e^{-\left(\frac{x^2}{2\sigma^2} + \frac{y^2}{2\sigma^2}\right)}, \quad (2.19)$$

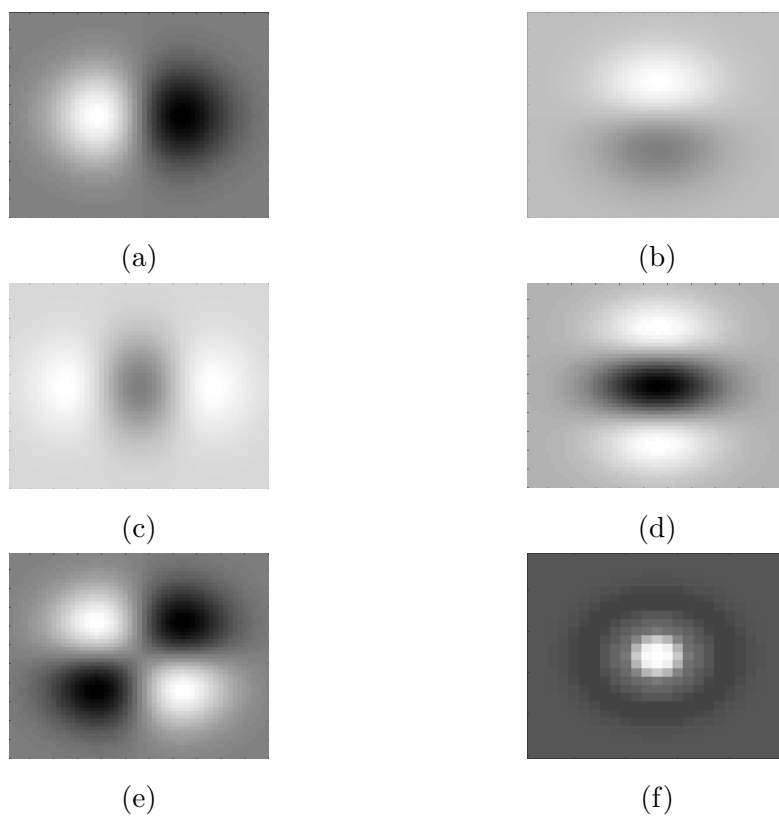
filter $G_{\sigma K}$ pa je enak:

$$G_{\sigma K}(x, y) = e^{-\left(\frac{x^2}{2\sigma^2 K^2} + \frac{y^2}{2\sigma^2 K^2}\right)}. \quad (2.20)$$

Zaradi lastnosti konvolucije pa velja tudi:

$$C_{DoG} = I * (G_{\sigma} - G_{\sigma K}). \quad (2.21)$$

V enačbah (2.18) in (2.21) je σ standardni odklon, K pa faktor za katerega se standardna odklona obeh filtrov razlikujeta. Razliko dveh filtrov z različnimi σ uporabimo tudi pri usmerljivih filtrih predlagane metode **F.2**, **F.3** in **F.5**. Seštevanje in odštevanje jeder filtrov je mogoče zaradi distributivne lastnosti konvolucije. Na Sliki 2.2 so prikazana posamezna konvolucijska jedra obravnavanih filtrov.



Slika 2.2: Konvolucijska jedra za uporabljene filtre. a) Konvolucijsko jedro prvega odvoda Gaussove funkcije po x. b) Konvolucijsko jedro prvega odvoda Gaussove funkcije po y. c) Konvolucijsko jedro drugega odvoda Gaussove funkcije po x. d) Konvolucijsko jedro drugega odvoda Gaussove funkcije po y. e) Konvolucijsko jedro drugega odvoda Gaussove funkcije po x in y. f) Konvolucijsko jedro DoG.

2.2 Radialnost

Radialnost [31] je način detekcije interesnih točk, ki bazira na samopodobnosti regij v sliki. Regija je samopodobna v primeru, da velja:

$$I(T(x)) = a + bI(x), \forall x \in P, \quad (2.22)$$

kjer je P krožna regija radija R , \mathbf{x} pa je točka v P . T označuje bijektivno geometrično transformacijo definirano na P kot sta rotacija okoli središča ali pa zrcaljenje čez črto, ki potuje skozi center P . $I(\mathbf{x})$ pa je svetlost slike v točki x .

Rotacija okoli centra P za kot α preslika točko $\mathbf{x} = [r, \phi]^T$, zapisano v polarnih koordinatah, v točko $[r, \phi + \alpha]^T$ kjer je r radij od središča P , ϕ pa kot. Zrcaljenje čez črto z orientacijo ϑ pa točko $\mathbf{x} = [r, \phi]^T$ preslika v $[r, 2\vartheta - \phi]^T$.

Z uporabo realnih podatkov je enačba (2.22) izpolnjena le v redkih primerih vendar pa lahko samopodobnost posamezne regije glede na transformacijo T izračunamo tudi z normaliziranim korelacijskim koeficientom:

$$nkk(P, T) = \frac{\sum_i (I(\mathbf{x}_i) - \bar{I}) \sum_i (I(T(\mathbf{x}_i)) - \bar{I})}{\sum_i (I(\mathbf{x}_i) - \bar{I})^2}. \quad (2.23)$$

V enačbi (2.23) x_i predstavlja posamezno točko P , \bar{I} pa predstavlja povprečno intenzivnost regije P . Zaradi poenostavitve računanja vsako regijo P najprej pretvorimo iz kartezijskih v polarne koordinate. To pretvori regijo P iz okrogle regije z radijem R v pravokotno regijo $P(M, N)$ kjer M in N označujeta število vzorčenj za različne radije (M) in različne kote (N). Tako se glede na željeno natančnost v regijah za kot ϕ in radij r vzorči po intervalih $\Delta\phi = \frac{2\pi}{N}$ ter $\Delta r = \frac{R}{M}$.

Če je transformacija T zrcaljenje moramo za izračun radialnosti izračunati normalizirani korelacijski koeficient za vsako od N vzorčenih orientacij ϑ . V primeru, da je T rotacija pa moramo izračunati normalizirani korelacijski

koeficient za vsakega od N vzorčenih kotov α . Ne glede na to ali je T rotacija ali zrcaljenje je radialnost enaka. Radialnost S_r je:

$$S_r = \frac{1}{N} \frac{1}{V_P} \sum_{m=0}^{M-1} (C_m - \bar{C})^2, \quad (2.24)$$

kjer je V_P vsota kvadratov, \bar{I} pa povprečna vrednost vseh vrednosti P :

$$V_P = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} (I(r_m, \phi_n) - \bar{I})^2, \quad (2.25)$$

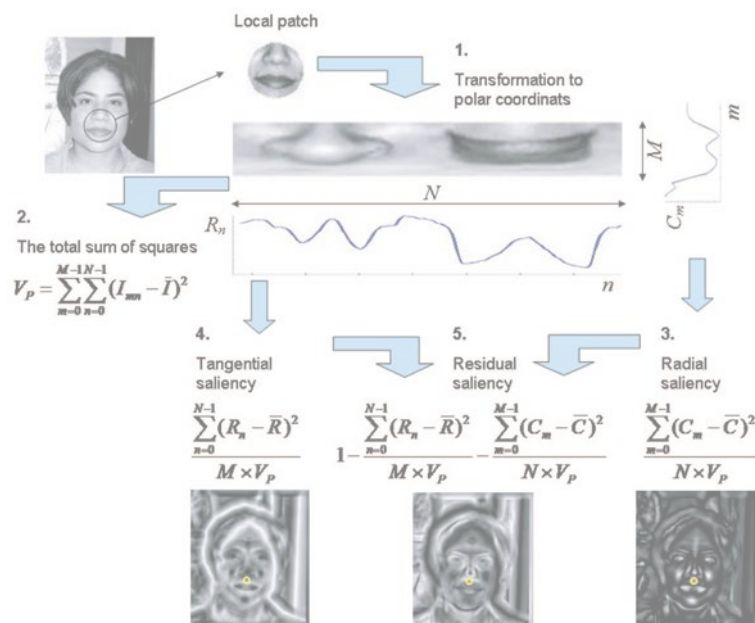
$$\bar{I} = \frac{\sum_{m=0}^{M-1} \sum_{n=0}^{N-1} I(r_m, \phi_n)}{NxM}, \quad (2.26)$$

C_m pa je vsota vseh vrednosti P na radiju r_m :

$$C_m = \sum_{n=0}^{N-1} I(r_m, \phi_n), \quad (2.27)$$

$$\bar{C} = \frac{1}{M} \sum_{m=0}^{M-1} C_m. \quad (2.28)$$

Slika 2.3 povzema postopek računanja radialnosti. S številko 3 je označena radialnost, s številkami 4 in 5 pa meri, ki ju v sklopu tega dela ne obravnavamo.



Slika 2.3: Postopek računanja radialnosti. Slika vzeta iz [31].

2.3 Segmentacija Felzenszwalb

Algoritem [16] sta ga v letu 2004 popularizirala Felzenszwalb in Huttenlocher. Je eden izmed popularnejših algoritmov za segmentacijo slik v regije. Poenostavljena ideja algoritma je združevanje med seboj podobnih manjših regij v večje in tako generirati predloge za interesne regije v slikah. Postopek za generiranje predlogov regij [16] je sledeč. Najprej se na sliki izračuna minimalno vpeto drevo (MST) z uporabo Kruskalovega algoritma [25]. Na sivinski sliki lahko za težo posamezne povezave v grafu $w((v_i, v_j))$, ki povezuje točki v_i in v_j uporabimo enačbo:

$$w((v_i, v_j)) = |I(p_i) - I(p_j)|. \quad (2.29)$$

Točki v_i in v_j predstavljata posamezne piksele. Po računanju MST se sliko zgladi z Gaussovim filtrom z manjšim parametrom σ . Avtorja predla-

gata $\sigma = 0.8$. Sosednje točke v grafu, ki predstavljajo začetne komponente, se združuje v večje komponente glede na kriterije o podobnosti znotraj regije ter med regijami.

Kriterij $Dif(\Gamma_x, \Gamma_y)$ je torej enak najnižji uteži povezave med dvema komponentama Γ_x in Γ_y in ga izračunamo po enačbi:

$$Dif(\Gamma_x, \Gamma_y) = \min_{v_i \in X, v_j \in Y, (v_i, v_j) \in E} w(v_i, v_j), \quad (2.30)$$

kjer $Dif(\Gamma_x, \Gamma_y)$ označuje kriterij podobnosti med komponentama Γ_x in Γ_y . \mathbf{v}_i je točka v komponenti Γ_x , \mathbf{v}_j pa je točka v komponenti Γ_y . Povezava med točkami (v_i, v_j) je vzeta iz nabora povezav MST E .

Kriterij podobnosti znotraj komponente Γ_x izračunamo po enačbi:

$$Int(\Gamma_x) = \min_{e \in E} w(e), \quad (2.31)$$

kjer je $Int(\Gamma_x)$ vrednost podobnosti znotraj komponente Γ_x , ki je enaka najnižji uteži $w(e)$ na povezavi e množice povezav E . E je množica vseh povezav znotraj MST komponente Γ_x . Za odločanje ali se regiji združita ali ne uporabimo združevalni pogoj:

$$Dif(\Gamma_x, \Gamma_y) \leq \min\left\{Int(\Gamma_x) + \frac{k}{|\Gamma_x|}, Int(\Gamma_y) + \frac{k}{|\Gamma_y|}\right\}. \quad (2.32)$$

Enačba (2.32) je odvisna od velikosti regije $|\Gamma_x|$ in $|\Gamma_y|$ ter parametra k , ki je nek konstanten faktor, ki prisili manjše regije v združevanje saj je pri zelo majhnih komponentah $Int(X)$ slab pokazatelj podobnosti znotraj komponente in bi bilo torej primerjanje z $Dif(X, Y)$ nenatančno.

Algoritem 1 povzema postopek segmentacije [16]. Graf $G = (V, E)$ predstavlja MST vhodne slike. V je množica n točk in E množica m povezav.

Pri obdelavi barvnih slik se algoritem izvede za vsak barvni kanal nato pa rezultate združimo. Sosednja piksla v barvni sliki dodelimo isti komponenti

Razvrsti povezave E po nepadajočih vrednosti $w(v_i, v_j)$.

Začnemo s segmentacijo S^0 , kjer je vsaka točka svoja komponenta.

```

for  $q = 1, \dots, m$  do
    Naj bosta  $v_i$  in  $v_j$  točki, ki ju povezuje povezava  $e_q$ .
    Naj bosta  $C_i$  in  $C_j$  komponenti točk  $v_i$  in  $v_j$ 
    if Če sta  $v_i$  in  $v_j$  v različnih komponentah trenutne segmentacije
     $S^{q-1}$  then
        if Drži formula 2.32 then
            Združi komponenti  $C_i$  in  $C_j$ .
             $S^q = ( S^{q-1}$  po združitvi  $C_i$  in  $C_j$  )
        end
    else
         $S^q = S^{q-1}$ 
    end
end

```

Algorithm 1: Segmentacijski algoritem [16]

le, če pripadata isti komponenti v rezultatu vseh treh kanalov.

2.4 Selective Search

Glede na pregled metod [21] je Selective Search[46] eden izmed boljših algoritmov za predlaganje regij. Glavna ideja algoritma je iterativno združevanje regij začetne segmentacije glede na več različnih mer podobnosti regij. Začetna segmentacija na manjše regije se generira z algoritmom [16]. Za vsak par sosednjih regij moramo najprej izračunati podobnost med njima glede na mere podobnosti, ki so opisane v Poglavju 2.4.1. Regije z najvišjo medsebojno podobnostjo se nato združuje v večje regije hkrati pa se posodablja podobnosti med novo nastalimi regijami ter njihovimi sosedi. Algoritem se konča, ko so vse regije začetne segmentacije združene v eno. V nadaljevanju so opisane mere podobnosti na podlagi katerih se posamezne komponente združujejo.

2.4.1 Mere podobnosti

Regije so si med sabo podobne če izpolnjujejo kriterije o podobnosti barve, teksture, velikosti ali oblike.

Barvna podobnost Na vsaki regiji R_i izračunamo enodimenzionalni histogram za vsak barvni kanal v regiji, ki jih nato združimo v enodimenzionalni histogram C_i , ki vsebuje histograme vseh barvnih kanalov. V delu [46] se predlaga uporaba 25 celic v vsakem histogramu barvnega kanala. To pripelje do barvnega histograma $C_i = \{c_i^1, \dots, c_i^n\}$, kjer je n dimenzionalnost histograma. V RGB barvnem prostoru, ki ima 3 barvne kanale je tako dimenzionalnost histograma $n = 75$. Barvno podobnost regij R_i in R_j izračunamo s presekom obeh histogramov:

$$S_b(R_i, R_j) = \sum_{k=1}^n \min(c_i^k, c_j^k), \quad (2.33)$$

kjer sta c_i^k in c_j^k vrednosti v histogramih C_i in C_j regij R_i in R_j .

Teksturna podobnost Teksture algoritem predstavlja z meritvami podobnimi SIFT-u [28] oziroma histogramu orientiranih gradientov [10]. Za vsak barvni kanal slike se izračuna Gaussov odvod v osmih orientacijah.

Za vsako orientacijo se konstruira histogram z desetimi celicami. Tako na primer za regijo R_i na sliki predstavljeni v RGB prostoru pridobimo 240 dimenzionalni deskriptor T_i . T_i lahko predstavimo z $T_i = \{t_i^1, \dots, t_i^n\}$, kjer je n dimenzionalnost T_i . Teksturno podobnost med regijami tako izračunamo u enačbo:

$$S_t(R_i, R_j) = \sum_{k=1}^n \min(t_i^k, t_j^k), \quad (2.34)$$

kjer sta t_i^k in t_j^k vrednosti v deskriptorjih T_i in T_j regij R_i in R_j .

Podobnost v velikosti Kot metoda opisana v Poglavju 2.3, ima tudi Selective Search mehanizem za spodbujanje zgošnjega združenja majhnih regij. To pripomore k temu, da so regije med potekom algoritma bolj enakih velikosti. Podobnost v velikosti je definirana kot delež slike I , ki ga skupaj pokrivata R_i in R_j :

$$S_v(R_i, R_j) = 1 - \frac{\Omega(R_i) + \Omega(R_j)}{\Omega(I)}, \quad (2.35)$$

kjer sta $\Omega(R_i)$ in $\Omega(R_j)$ velikosti regij R_i in R_j , $\Omega(I)$ pa je velikost slike I .

Podobnost v obliki Merimo lahko tudi kako dobro se regiji druga drugi prilagata. Kot primer vzemimo regiji, kjer je ena vgnezdena v drugi. Smiselno bi bilo, da bi regiji v naslednjem koraku združili saj z večjo verjetnostjo predstavljata isti predmet. V nasprotnem primeru, če se regiji komaj stikata je verjetnost, da združeni predstavljata isti predmet, majhna. Regijama R_i in R_j očrtamo pravokotnik, ki vsebuje obe regiji. Podobnost v obliki definiramo kot delež slike I , ki ga predstavlja del očrtanega pravokotnika, ki ne vključuje dela slike, ki ga prekrivata regiji:

$$S_o(R_i, R_j) = 1 - \frac{\Omega(BB_{ij}) - \Omega(R_i) - \Omega(R_j)}{\Omega(I)}, \quad (2.36)$$

kjer je BB_{ij} regija, ki jo predstavlja pravokotnik očrtan regijama R_i in R_j . Z Ω so označene velikosti regij.

Kombinirana podobnost Zgoraj našteje mere podobnosti se lahko tudi združuje s čimer dobimo novo mero podobnosti, ki upošteva različne lastnosti regij. Kombinirano podobnost izračunamo kot linearno kombinacijo ostalih kjer so a_1, a_2, a_3 in a_4 koeficienti s katerimi določamo kakšen vpliv ima posamezna meritev na vrednost kombinacije:

$$S(R_i, R_j) = a_1 S_b(R_i, R_j) + a_2 S_t(R_i, R_j) + a_3 S_v(R_i, R_j) + a_4 S_o(R_i, R_j). \quad (2.37)$$

Algoritem se izvaja tudi v različnih barvnih prostorih saj lahko zaradi lastnosti posameznega barvnega prostora dobimo različne rezultate. V barvnem prostoru RGB spremembe v osvetlitvi vplivajo na vse tri barvne kanale [43]. V prostoru HSV sprememba v osvetlitvi ne vpliva na kanal H, S se ne spreminja s sencami, V pa ni občutljiv na spremembe v svetlosti. Prostor Lab pa uporablja kanal svetlosti in dva barvna kanala (a in b). Podobno kot HSV je neobčutljiv na spremembe v sencah ali svetlosti. Pri izvajanju algoritma se uporablja tudi sivinsko sliko.

2.5 Algoritem ACF

Algoritem ACF (ang. Aggregated Channel Features) je bil predlagan v delu [11]. ACF deluje tako, da na sliki najprej izračuna tri kanale. Za kanale uporabi histogram orientiranih gradientov (HOG) [10], normalizirano magnitudo gradienta in barvne kanale LUV. Gradient slike I ter njegova orientacija sta podana v enačbah (2.11) ter (2.12). Magnitudo gradienta pa izračuna z enačbo:

$$|\Delta I| = \sqrt{\left(\frac{\delta I}{\delta x}\right)^2 + \left(\frac{\delta I}{\delta y}\right)^2}. \quad (2.38)$$

Normalizirano magnitudo gradienta M_{norm} izračunamo za vsak del slike, ki jo predhodno razdelimo na prostore v velikosti 11 x 11 pikslov. Za izračun normalizirane magnitude gradienta potrebujemo tudi povprečno magnitudo gradienta vsakega dela slike \overline{M} , ki jo pridobimo s konvolucijo magnitude gradienta $|\Delta I|$ z l_1 normaliziranim trikotnim filtrom velikosti 11 x 11 pikslov [41]. M_{norm} izračunamo z enačbo:

$$M_{norm} = \frac{M}{\overline{M} + 0.005}. \quad (2.39)$$

Algoritem ACF uporablja tudi deskriptor HOG [10], ki v metodi predstavlja n kanalov. Parameter n je nastavljen in opisuje v koliko različnih smereh se bo izračunal gradient slike. V delu [11] je predstavljen pristop s šestimi kanali HOG. Sliko se razdeli v celice nato pa se z vsemi piksli v celici izdelava histogram orientacij glede na vrednosti gradientov. Zaradi razlik v osvetlitvi in kontrastu je potreben še normalizacijski korak. Celice se najprej združijo v večje povezane skupine. Avtorja algoritma HOG predlagata uporabo štirih 8 x 8 pikslov velikih celic za sestavo ene povezane skupine. Vsako skupino se nato normalizira z eno od metod za normalizacijo. Algoritem ACF za detekcijo kanale aproksimira za detekcijo značilnic različnih velikosti.

Zadnji kanal metode ACF je barvni kanal LUV. V delu [11] odločitev za izbor LUV kot kanal algoritma ACF ni obrazložena vendar pa so avtorji članka v delu [13] tudi z uporabo LUV dosegli dobre rezultate pri detekciji ljudi na slikah. Algoritem ACF uporablja enake kanale kot delo [12].

Po generiranju vseh kanalov slike se kanale razdeli v celice velike 4 x 4 piksele. Vrednosti v posamezni celici se nato sešteje s čimer dobimo kanale v veliko nižji resoluciji, ki se jih kasneje vektorizira. Te značilke se nato uporabi v algoritmu AdaBoost [18], ki je poznan predvsem po detekciji obrazov [47, 48]. Temelji na metodi Boosting [39], katere ideja je grajenje močnega klasifikatorja iz zaporedja šibkih uteženih klasifikatorjev, ki pa so za dani problem boljši od naključnega izbiranja razreda. Recimo, da imamo za učenje

algoritma na voljo N primerov (x_i, y_i) od česar je L pozitivnih ter M negativnih primerov. x_i označuje dani primer, y_i pa označuje klasifikacijo primera $y_i = 0, 1$. Algoritem najprej inicializira uteži za vsak primer z vrednostjo $\frac{1}{M+L}$ [40]:

$$w(i) = \frac{1}{M+L}, \quad (2.40)$$

nato za vsako iteracijo t uteži najprej normaliziramo, da pridobimo verjetnostno porazdelitev w_t :

$$w_t(i) = \frac{w(i)}{\sum_{j=1}^n w(j)}. \quad (2.41)$$

Za vsako značilko j natreniramo klasifikator h_j ter izberemo tistega, ki doseže najnižjo klasifikacijsko napako e_t ter z uporabo napake posodobimo uteži. Napako izračunamo z enačbo:

$$e_t = \sum_i w_i |h_j(x_i) - y_i|. \quad (2.42)$$

Po izboru klasifikatorja h_t z minimalno napako e_t posodobimo uteži tako, da imajo pravilno klasificirani primeri nižjo težo v nadaljevanju algoritma kot pa nepravilno uvrščeni. Uteži posodabljam po formuli:

$$w_{t+1}(i) = w_t(i) \beta_t^{1-e_i}. \quad (2.43)$$

V zgornji formuli je $\beta_t = \frac{e_t}{1-e_t}$. Velja $e_i = 0$ če je x_i klasificiran pravilno ter $e_i = 1$ v primeru, da je bil primer napačno klasificiran. Zgornje korake ponovimo za vsak t v T iteracijah nato pa pridobimo končni klasifikator h :

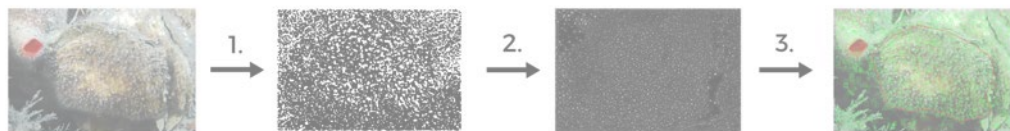
$$h(x) = \begin{cases} 1, & \sum_{t=1}^T \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^T \alpha_t \\ 0, & \text{ostalo} \end{cases}.$$

V algoritmu ACF se AdaBoost uporablja za treniranje in kombiniranje odločitvenih dreves [36] s prej pridobljenimi značilkami. V članku avtorji predlagajo uporabo 2048 odločitvenih dreves z globino dve.

Poglavje 3

Predlagana metoda

Predlagana metoda, ki jo v nadaljevanju imenujemo metoda detekcije radialnih regij (RRD), za generiranje hipotetičnih regij za detekcijo polipov deluje v dveh delih. Najprej se z uporabo kombinacije usmerljivih filtrov [17] generira masko, ki pokriva lokacije polipov na sliki, nato pa se izračuna masko uporabi pri računanju radialnosti [31] s pomočjo katere določimo natančnejšo lokacijo in velikost za predlagane regije. V Poglavlju 3.1 opisujemo postopek uporabljen za generiranje maske, v Poglavlju 3.2 pa generiranje hipotetičnih regij iz maske. Na Sliki 3.1 je predstavljen potek predlagane metode.



Slika 3.1: V 1. koraku metode se izvede generiranje maske. V 2. koraku s pomočjo generirane maske izračunamo radialnost na sliki [31]. V 3. koraku pa z izračunom lokalnih maksimumov radialnosti predlagamo hipotetične regije polipov.

3.1 Generiranje maske

Za generiranje maske izkoriščamo vizualne lastnosti polipov. Postavimo predpostavko, da so polipi majhne, okrogle in svetle regije, ki iz svojega ozadja po svetlosti izstopajo. Kot vidimo na Sliki 3.2, to pogosto drži. Za detekcijo okroglih svetlih struktur uporabimo filter DoG [27], ki je popularen detektor okroglih regij (ang. blob detector) [26]. Za izboljšanje diskriminativne moči maske uporabimo še več različnih filtrov, ki imajo višje odzive na drugačne strukture v sliki kot DoG. Na Sliki 3.3 je prikazan potek generiranja maske.



Slika 3.2: Polipi so velikokrat okrogli in se po svetlosti razlikujejo od ozadja.

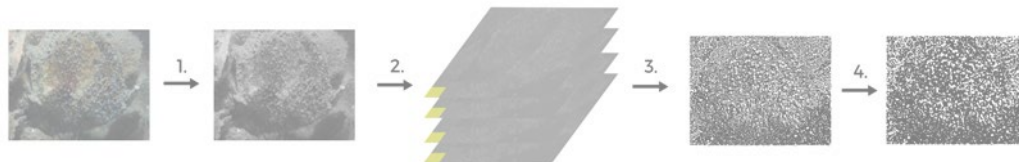
V prvem koraku najprej vhodno sliko pretvorimo v sivinsko sliko I , ki jo nato obdelamo z vsakim izmed filtrov g_i iz nabora filtrov G :

$$f_i(p) = [I * g_i](p), \quad (3.1)$$

kjer je p piksel v sliki I , $*$ pa operator konvolucije. Ob generiranju maske poskušamo vsak piksel klasificirati v teksturni razred, ki ga opisuje vsak izmed filtrov g_i . Klasifikacijska funkcija je:

$$C(p) = \arg \max_i f_i(p), \quad (3.2)$$

ki piksel p klasificira v razred i v primeru, da ima na danem pikslu najvišji



Slika 3.3: V koraku označenem s številom 1 sliko pretvorimo v sivinsko sliko. V 2. koraku sivinsko sliko obdelamo z vsakim filtrom v naboru ter za vsak piksel poiščemo maksimalno vrednost. V 3. koraku obdržimo le piksele, kjer maksimalen odziv poda DoG. V zadnjem koraku pa se na maski pridobljeni v 3. koraku izvedejo morfološke operacije.

odziv prav filter g_i . Označimo filter DoG z g_1 . Generiranje maske $h(p)$ s pomočjo klasifikacijske funkcije $C(p)$ zapišemo s formulo:

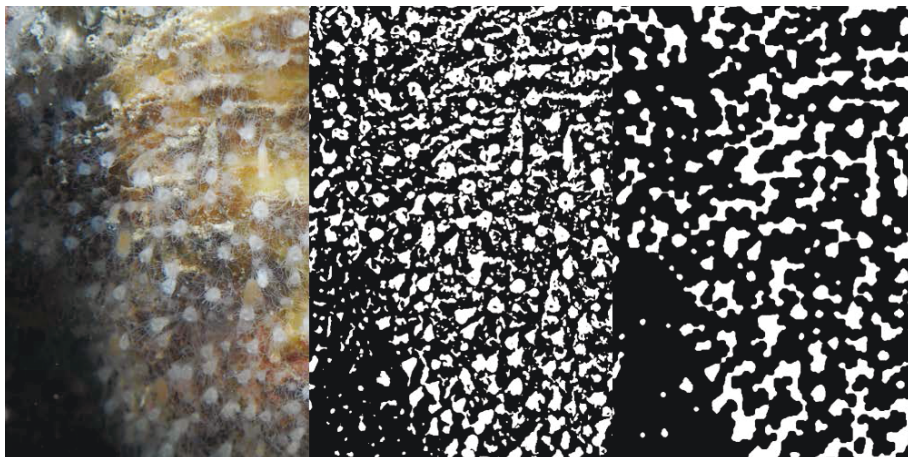
$$h(p) = \begin{cases} 1, & C(p) \equiv 1 \\ 0, & \text{ostalo} \end{cases}.$$

Funkcija $h(p)$ generira binarno masko I_m in piksel p doda v masko le v primeru, da funkcija $C(p)$ piksel klasificira v razred 1 za kar pa mora maksimalno vrednost za p podati $f_1(p)$, ki pa predstavlja vrednost rezultata konvolucije s filtrom DoG. V naboru filtrov G je pet filtrov naštetih v Poglavju 2.1.

3.1.1 Morfološke operacije

Na maski I_m se vedno pojavlja določena količina šuma, ki včasih negativno vpliva na natančnost generiranja regij in velikokrat povzroči previsoko raven segmentacije na polipih. Zaradi spremembe intenzivnosti znotraj polipa zaradi njegove delne prosojnosti lahko torej algoritem posamezen polip segmentira v več regij ali pa zaradi raznolikosti ozadja predlaga veliko nepravilnih regij, ki ne vsebujejo polipov. Problem šuma smo reševali s morfološkimi operacijami na maski. Za odstranjevanje nastalega šuma smo uporabili mor-

fološki operaciji odpiranja in zapiranja. Na Sliki 3.4 je prikazan primer maske z uporabo in brez uporabe morfoloških operacij.



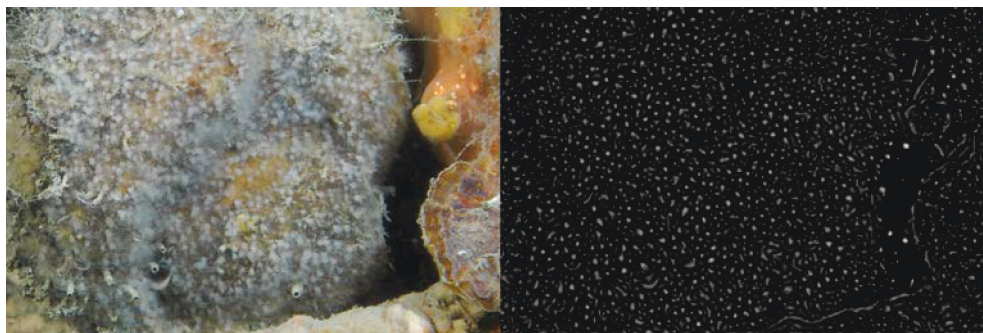
Slika 3.4: Od leve proti desni: neobdelana slika polipov, generirana maska brez uporabe morfoloških operacij, generirana maska z uporabo morfoloških operacij.

Zaradi možnosti uspeha metode brez uporabe morfoloških operacij smo se odločili v naše eksperimentalno delo dodati tudi to verzijo predlagane metode.

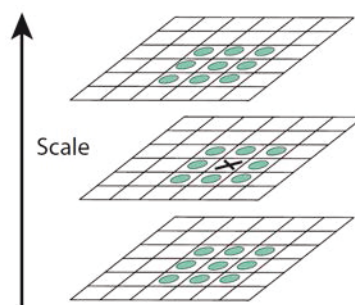
3.2 Predlaganje regij

Radialnost se v metodi izračuna na celotni sliki polipov ter maski generirani v prvem delu. Računa se za vsak radij r_m zaradi česar za posamezno sliko dobimo M nivojski zemljevid radialnosti, kjer vsak nivo odraža radialnosti struktur na sliki pri velikosti r_m . Primer nivoja zemljevida radialnosti je prikazan na Sliki 3.5.

Regije, ki jih algoritem predlaga, so postavljene na pozicije lokalnih ekstremov. Postopek iskanja lokalnih maksimumov poteka tako, da za vsak



Slika 3.5: Od leve proti desni: Neobdelana slika polipov, primer nivoja zemljevida radialnosti na sliki polipov.

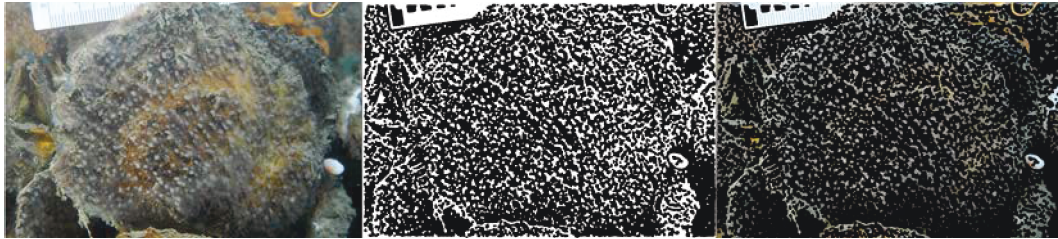


Slika 3.6: Prikaz primerjave piksla, na sliki označenega z x , z osmimi sosednjimi piksi na njegovem nivoju ter devetimi sosednjimi piksi na višjem in nižjem nivoju. Slika pridobljena iz [29].

radij r_m vrednost vsakega piksla primerjamo z osmimi sosedmi na istem nivoju zemljevida radialnosti ter z devetimi sosedmi v nivoju višje in nižje na zemljevidu. Postopek primerjave je prikazan na Sliki 3.6. Regije predlagamo le na lokacijah kjer ima določen piksel na nivoju strogo višjo vrednost od svojih sosedov na njegovem kot tudi na višjem in nižjem nivoju. Predlagano metodo imenujemo detekcija radialnih regij in se bomo v nadaljevanju nanjo sklicevali kot RRD.

3.3 Kombinacija z algoritmom Felzenszwalb

Ob vizualni analizi maske, ki jo generira algoritem RRD smo opazili, da maska sicer dobro predstavlja lokacije polipov vendar pa so regije slabše segmentirane saj posamezna regija v maski velikokrat pokriva več polipov. Sliko polipov smo kombinirali z masko tako, da smo v sliki ohranili le piksele, ki so bili v maski predstavljeni s pozitivno vrednostjo. Maska, slika ter njuna kombinacija je prikazana na Sliki 3.7. Na tako maskirani sliki smo nato uporabili algoritem opisan v [16].



Slika 3.7: Od leve proti desni: neobdelana slika polipov, generirana maska, maskirana slika. Maskirano sliko pridobimo tako, da v nespremenjeni sliki zberemo vse piksele, ki so v generirani maski predstavljeni z nič.

Poglavje 4

Eksperimentalna analiza

V eksperimentalni evaluaciji smo se osredotočili na podrobnejše ocenjevanje metode RRD in na analizo posameznih delov metode z namenom optimizacije rezultatov pri problemu predlaganja regij za polipe. V Poglavju 4.1 je opisana podatkovna zbirka na kateri so se izvajali eksperimenti, v Poglavju 4.3 pa smo na zbirki uporabili metodo RRD in preučili vpliv eliminacije posameznih filtrov za generiranje maske na uspešnost predlaganja regij. Preizkušena je bila tudi kombinacija metode RRD in segmentacije Felzenszwalb [16] katere rezultati so predstavljeni v Poglavju 4.8. Rezultate predlagane metode smo predstavili v Poglavju 4.7 ter jih primerjali z metodo ACF [11] v Poglavju 4.5 in metodo Selective Search [46] v Poglavju 4.6. Ocenili smo tudi kombinacijo metod ACF in RRD ter rezultate opisali v Poglavju 4.9. V Poglavju 4.4 je podan pregled nad rezultati.

4.1 Podatkovna zbirka

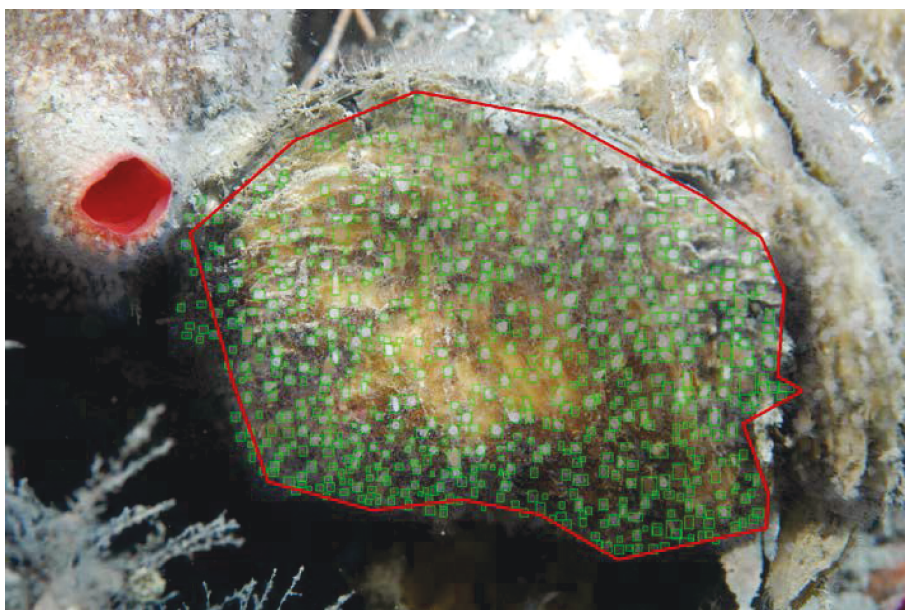
Podatkovna zbirka na kateri smo ocenjevali uspešnost posameznih metod je bila sestavljena iz 35 barvnih slik ostrig na katerih se nahajajo polipi. Vsaka slika je široka 4288 pikslov in visoka 2848 pikslov. Vsaki sliki je priložena tudi datoteka z ročno označenimi pravokotniki, ki v slikah obdajajo posamezne polipe. Del podatkovne zbirke so tudi podatki na katerem območju

	Širina	Višina	Diagonala
Povprečje	38	43	58
Standardni odklon	9.912	11.377	13.980
Maksimum	111	126	155
Minimum	11	12	18

Tabela 4.1: Podatki o anotacijah polipov v piksljih.

znotraj slike, so bili označeni polipi. Na Sliki 4.1 je območje označeno z rdečo črto. V eksperimentih uporabljamo le regije znotraj teh območij in s tem dobimo boljšo predstav o številu predlaganih regij glede na anotacije. V celotni zbirki je 37403 anotacij polipov vsaka slika pa ima v povprečju 1011 označenih pravokotnikov. Minimalno število označenih polipov na sliki je 378, maksimalno pa 1713. Ostale podatke o označbah polipov lahko najdemo v Tabeli 4.1, ki prikazuje podatke o višini, širini in diagonali ter njihova odstopanja.

Zaradi različne velikosti, orientacije in prosojnosti polipov je uspešno ročno določanje pozicij polipov lahko zahtevno in posledično se tudi v anotacijah lahko pojavljajo napake ali pa vsaj dvom glede pravilnosti označenega pravokotnika. Pravilno anotacijo še dodatno otežuje pogosto prekrivanje polipov na nekaterih slikah, kot je prikazano na Sliki 4.2 in zamegljenosti nekaterih delov določenih slik zaradi napak pri fotografiranju kot je razvidno iz Slike 4.3 na kateri so polipi v ospredju bistveno manj zamegljeni v primerjavi z bolj oddaljenimi.



Slika 4.1: Na sliki ostrig so z zelenimi pravokotniki ročno označene pravilne lokacije polipov. Rdeča črta označuje območje v katerem so bili polipi označeni.



Slika 4.2: Območje na katerem so polipi tako gosto postavljeni, da je meje med njimi zelo težko natančno določiti. Drug drugega delno zakrivajo zato lahko rečemo, da je prikazano območje prekrivanja polipov.



Slika 4.3: Prikaz zamegljenosti slike

4.2 Ocenjevanje predlaganih regij

Za primerjavo je rezultat metod seveda potrebno oceniti. Za ocenjevanje posameznih generiranih regij smo uporabili ročne označbe polipov. Tudi pri metodah, ki predlagajo segmente različnih oblik kot sta RRD, ki predlaga okrogle regije ter Felzenszwalb, ki predlaga različne oblike segmentov, smo posamezne regije obdali s pravokotnikom za lažjo primerjavo z ročnimi anotacijami. Za ocenjevalni kriterij posamezne regije je bil uporabljen IOU (ang. intersection over union). Glede na [21] je regija z višjim razmerjem med presekom in unijo za kasnejšo klasifikacijo objekta boljša. Za vsako predlagano regijo smo preverili kakšno razmerje IOU ima z ročnimi anotacijami s katerimi se vsaj delno prekriva. Izračunali smo priklic za več različnih mejnih vrednosti IOU T kjer smo predlagano regijo šteli kot pravilno le če je pri preseku z neko ročno anotacijo dosegala vrednost IOU višjo od T .

Priklic R je v našem primeru definiran kot:

T = izbrana meja IOU, kjer predlagano regijo še štejemo kot pravilno.

O = množica ročno označenih regij

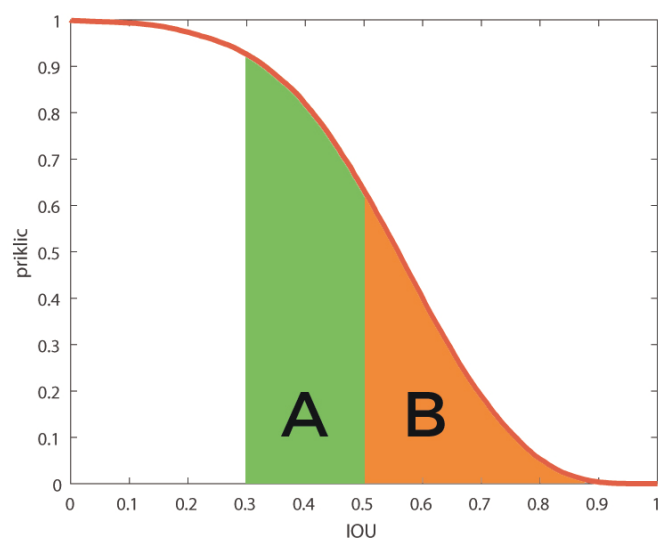
P = množica predlaganih regij z vrednostjo IOU več kot T

$$R = \frac{(O \cap P)}{O}. \quad (4.1)$$

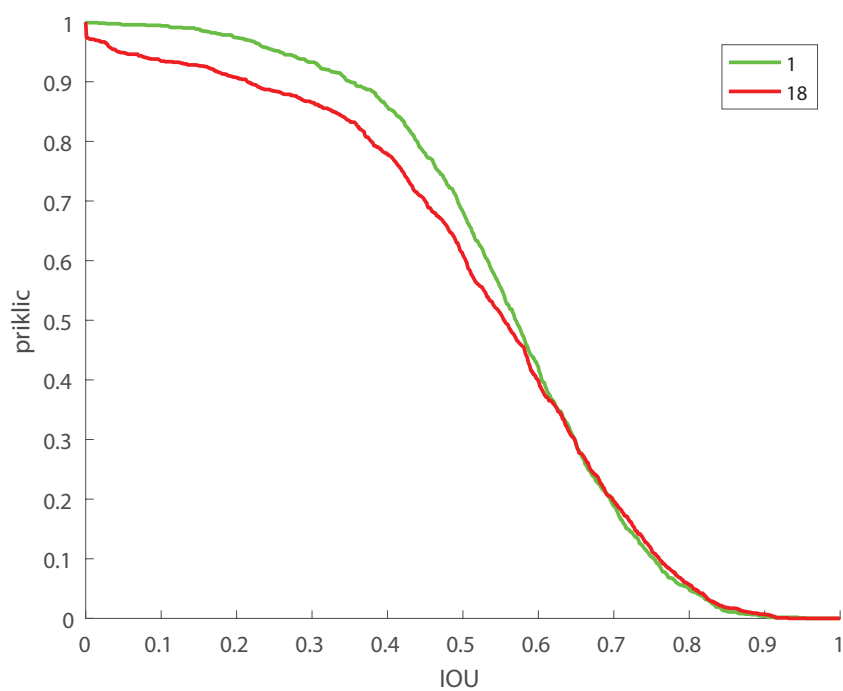
Priklic R je torej definiran kot delež števila ročno označenih regij O , ki jih prekriva vsaj ena od predlaganih regij P , kjer je stopnja prekrivanja IOU vsaj T .

Za splošno uspešnost posamezne metode na sliki je bila uporabljena površina pod krivuljo na grafu odvisnosti senzitivnosti od postavljene meje IOU za katero smo še šteli, da je pokritost posamezne anotacije uspešna. Za ocenjevanje uspešnosti smo sprva uporabili mero AR predlagano v [21], ki pri ocenjevanju upošteva le regije z IOU vrednostjo višjo od 0.5. Izkazalo se je, da delež predlaganih regij z IOU vrednostjo večjo od meje T začne hitro upadati, ko T pomikamo višje od 0.5. Delež predlaganih regij, ki še dosegajo

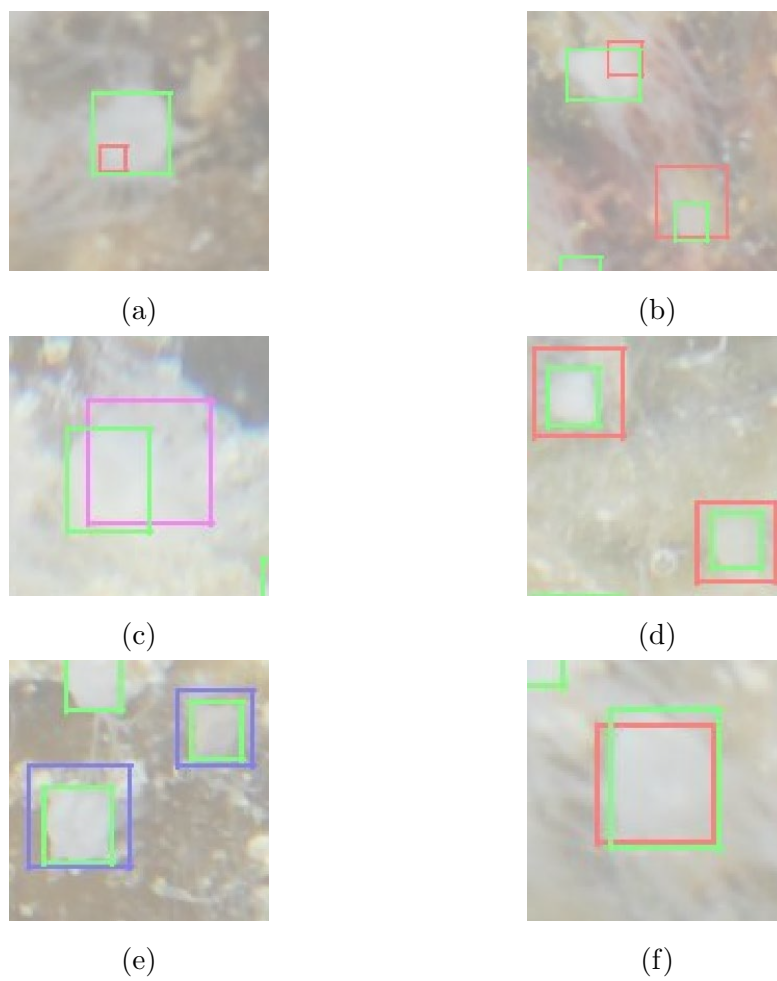
željeno vrednost IOU, začne z naraščajočo vrednostjo T hitro upadati, ko meja T preseže vrednost 0.5 zaradi česar je mera AR neprimerna za ocenjevanje pristopov k našemu problemu. Za ocenjevanje smo zato uporabili prilagojeno verzijo AR, ki prav tako uporablja površino pod krivuljo na grafu priklica v odvisnosti od IOU, le da so meje za ocenjen IOU predstavljene saj ne upošteva površine med IOU vrednostjo 0.5 in 1 temveč med 0.3 in 1. Grafična predstavitev posameznih mer je prikazana v Sliki 4.4. Nova mera omogoča boljše ocenjevanje pristopov zaradi konvergiranja uspešnosti pristopov z naraščajočo mejo T . Problem je prikazan na Sliki 4.5, kjer sta prikazani uspešnosti izvajanje metode RRD z različnimi parametri. Meja IOU med 0.3 ter 1 je smiselna tudi glede na vizualno analizo prekrivanja predlaganih regij z ročnimi anotacijami prikazanimi v Sliki 4.6. Regije z vrednostjo IOU manjšo od 0.3 slabo predstavljajo prave lokacije polipov, regije z višjim IOU pa jih že veliko bolje označujejo. Vrednosti posamezne regije so do neke mere odvisne tudi od anotacij saj so te lahko, zaradi lastnosti polipov, ki anotiranje otežujejo, nenatančne. Naša mera bo v nadaljnem označena kot AR3.



Slika 4.4: Primer grafa priklica v odvisnosti od IOU. Del površine pod krivuljo, na sliki označen z B, predstavlja mero AR. Celotno območje označeno z A in B predstavlja našo mero.



Slika 4.5: Kljub temu, da je uspešnost izvajanja metode, ki je označena z zeleno črto očitno boljša od tiste, prikazane z rdečo, se mera AR med njima skoraj ne razlikuje. V takem primeru je naša mera ocenjevanja boljša.



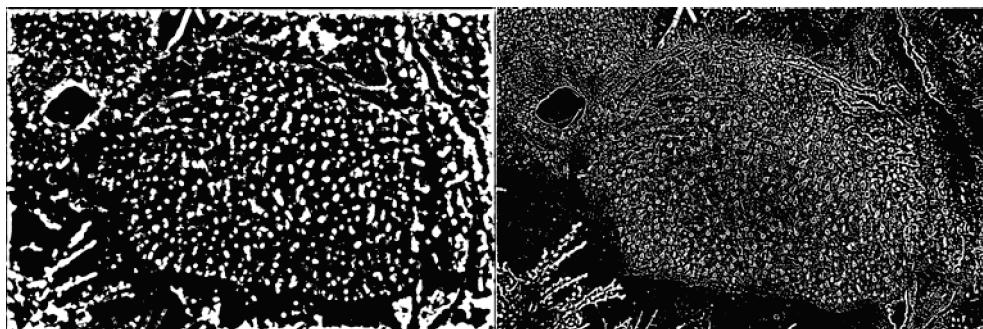
Slika 4.6: a) Prekrivanje anotacije in regije z $IOU = 0.1$. b) Prekrivanje z $IOU = 0.2$. c) Prekrivanje z $IOU = 0.3$. d) Prekrivanje z $IOU = 0.4$. e) Prekrivanje z $IOU = 0.5$. f) Prekrivanje z $IOU = 0.7$.

4.3 Izbira parametrov RRD

Za dobre rezultate je pri metodi RRD, kot tudi pri veliko drugih metodah, potreben izbor večih parametrov. Pri generiranju maske z usmerljivimi filtri je potrebna nastavitve parametrov σ in K , ki določata velikosti in obliko začetnih Gaussovih filtrov. Izkazalo se je tudi, da ima kombinacija in število izbranih usmerljivih filtrov velik vpliv na končen izgled maske in posledično tudi vpliv na predlagane regije. Problem izbora najboljših parametrov je veliko število kombinacij parametrov ter dejstvo, da parametri vplivajo na predlagane regije odvisno drug od drugega. Za bolj splošno obrazložitev rezultatov, neodvisno od podatkovne baze na kateri so se eksperimenti izvajali, bomo vrednosti parametrov, namesto v piksljih izražali v deležu diagonale povprečne ročne anotacije polipa. Povprečno diagonalo polipa, podano v Tabeli 4.1, bomo označevali kot λ_{diag} .

4.3.1 Izbor parametrov σ in K

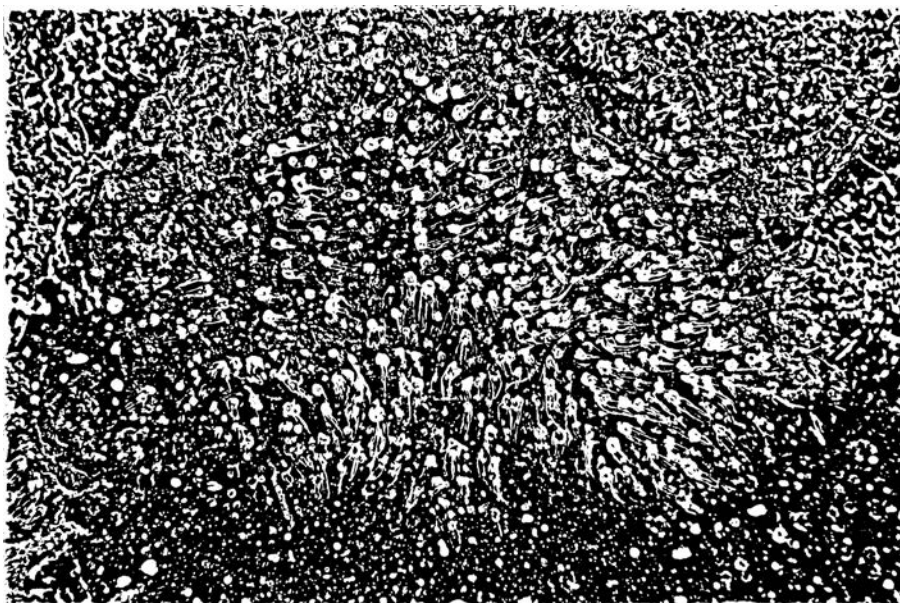
Za dobro segmentacijo slike ter natančno predstavitev lokacij polipov z uporabo metode RRD je potrebno generiranje dobre maske, ki pa je odvisna od nastavitve parametrov σ in K , ki jih uporabljamo v Gaussovih funkcijah v enačbi (2.3) ter pri sestavi filtrov DoG v enačbi (2.20). Vrednost parametrov smo določili z vizualno analizo, uspešnost pa kasneje še preverili z izvajanjem algoritma na generiranih maskah ter ocenjevanjem rezultata. Kot primer dobre vrednosti σ in K smo določili $\sigma = 0.05\lambda_{\text{diag}}$ in $K = 0.20\lambda_{\text{diag}}$. Podana vrednost je bila uporabljena na celotni podatkovni zbirki, kjer pa se velikost iskanih regij močno spreminja zato so parametri prilagojeni za delovanje na podani zbirki slik.



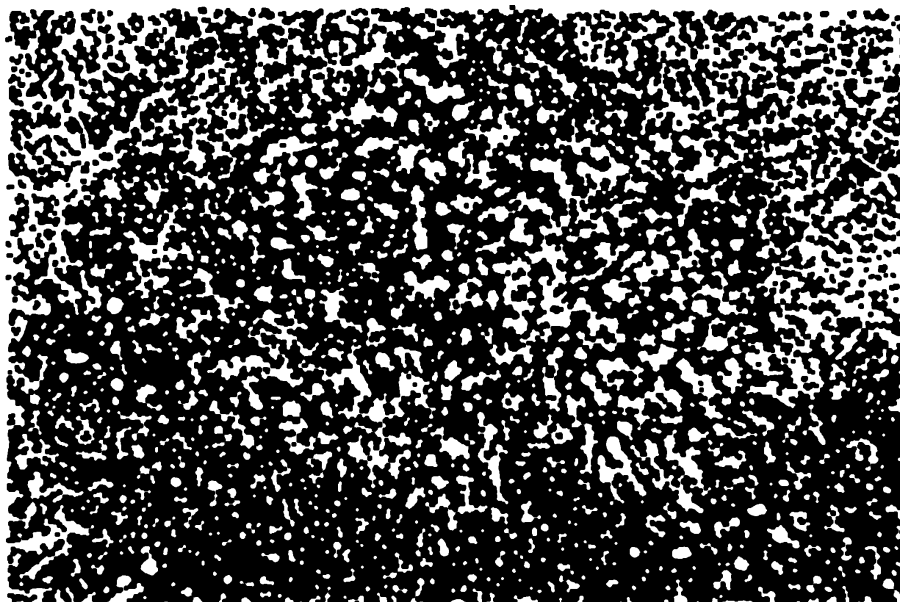
Slika 4.7: Na levi je primer pregrobe maske na desni pa primer preveč fine. Groba maska povzroči, da algoritem več regij polipov ne generira, fina pa generira preveč regij, ki slabo predstavljajo pozicije polipov.

4.3.2 Izbor morfoloških operacij

Sliki 4.9 in 4.8 prikazujeta generirani maski z enakimi parametri le, da smo prvo obdelali z operacijama odpiranja in zapiranja, drugo pa smo pustili neobdelano. Na Sliki 4.8 lahko vidimo, da čeprav algoritem dobro predstavi lokacije polipov vsebuje tudi veliko delov, ki so za detekcijo nesmiselni kot so lovke ter deli ozadja. Na Sliki 4.9 je veliko problematičnih delov odstranjenih, lahko pa opazimo, da je maska pri nekaterih polipih slabo razdeljena tako, da eno območje prekriva več polipov. To na rezultate ne vpliva tako negativno kot previsoka stopnja segmentacije pred morfološkimi operacijami. Vsako masko obdelamo najprej z odpiranjem, ki odstrani večino majhnih svetlih regij, ki polipov ne predstavljajo. Te nepravilne regije povzročijo elementi slike kot so lovke polipov, iregularnosti v ozadju ali manjši delci, ki plavajo v vodi. Po odpiranju sledi še obdelava maske z zapiranjem. S tem zapremo manjše luknje v regijah polipov, ki nastanejo zaradi spremembe v prosojnosti oziroma barvi posameznega polipa. Obe operaciji uporabljata okrogel strukturni element. Izbrane morfološke metode ter velikosti njihovih strukturnih elementov so bile določene z vizualno analizo mask po uporabi različnih kombinacij morfoloških operacij. Pri eksperimentih smo uporabili odpiranje in zapiranje z okroglim strukturnim elementom velikosti $0.1\lambda_{\text{diag}}$.



Slika 4.8: Maska pred apliciranjem operacij odpiranja in zapiranja. Opazimo lahko veliko šuma, ki ga povzročajo lovke in iregularnosti v okolju.



Slika 4.9: Maska po apliciranju operacij odpiranja in zapiranja.

4.3.3 Izbor kombinacije filtrov

Tudi izbor pravilne kombinacije filtrov v prvem koraku algoritma spada med pomembne parametre, saj se uspešnost metode na slikah konkretno spreminja glede na izbrano kombinacijo. Slika 4.10 prikazuje graf priklica v odvisnosti od IOU, ki je rezultat izvajanja metode z 31 različnimi kombinacijami filtrov na eni izmed slik. Sestavo posamezne kombinacije filtrov lahko razberemo iz Tabel 4.2 in 4.3. Vidimo lahko, da nekatere kombinacije dosežejo boljši rezultat kot druge, vendar pa je vredno omeniti, da nekatere kombinacije predlagajo bistveno manj regij kot druge. Že z vizualno analizo mask dveh kombinacij filtrov z različno stopnjo uspešnosti lahko iz čistosti maske napovemo, da bo ena izmed njiju predlagala manj regij na podoben način kot pred uporabo morfoloških operacij. Kombinacije, ki uporabljajo večje število filtrov večinoma generirajo maske z manj šuma kot tiste, ki upoštevajo le manjše število. Maske nastanejo tako, da na sliko apliciramo vse filtre in nato masko sestavimo iz pikslov kjer je odziv DoG maksimalen med vsemi odzivi filtrov na določenem pikslu. Tudi iz tega bi lahko sklepali, da je, če nekaj filtrov ne upoštevamo, večja verjetnost, da bo na neki lokaciji največji odziv izzval prav DoG.

Za uspešen izbor kombinacije je potrebna odločitev ali je pomembnejši priklic pri manjšem številu regij ali pri uporabi vseh generiranih lokacij polipov. Hitrost je za predvideno uporabo metode, štetje polipov, manj pomembna saj je glavni namen avtomatizacija štetja česar alternativa je ročno štetje, ki pa je drastično dolgotrajnejše. Zaradi tega razloga so v Poglavju 4.7 opisani rezultati metode izvajane z uporabo kombinacij, ki so v povprečju uspešne ne glede na število regij, ki jih predlagajo.

4.3.4 Parametri radialnosti

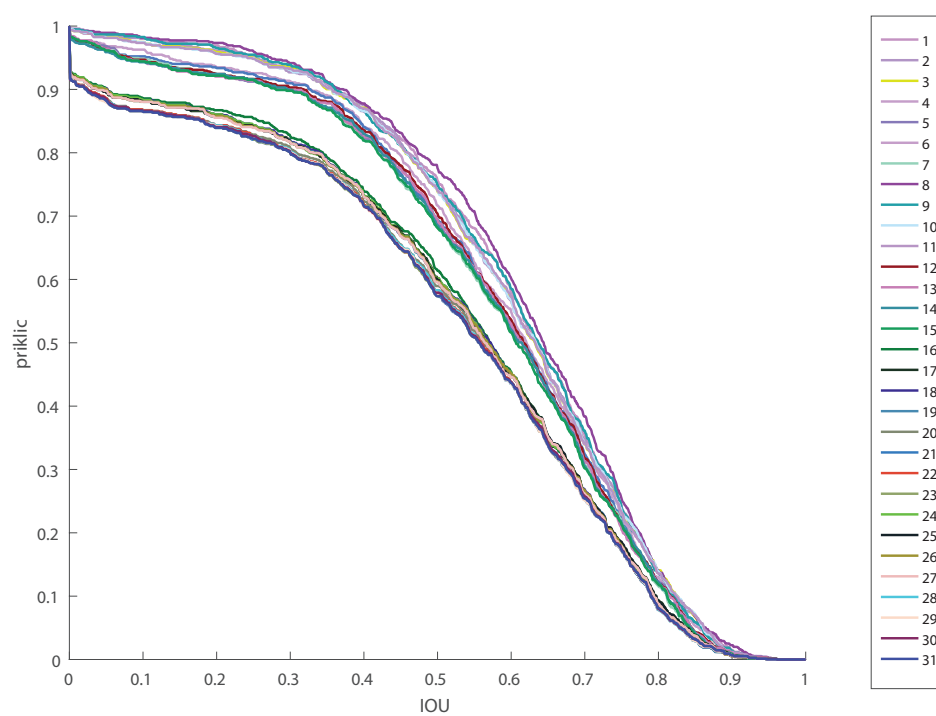
Ponovimo, da se v drugem koraku algoritma RRD s pomočjo maske in sive slike izračuna radialnost za vsakega od M radijev. Dobimo torej zemljevid radialnosti z M nivoji, ki nam omogoča ocenjevanje potencialne regije

Št. kombinacije	uporabljene metode
1	F.1, F.4
2	F.1, F.5
3	F.1, negativen F.3, F.4
4	F.1, F.5
5	F.1, F.5, F.4
6	F.1, F.5, negativen F.3
7	F.1, F.5, negativen F.3, F.4
8	F.1, F.3
9	F.1, F.3, F.4
10	F.1, F.3, negativen F.3
11	F.1, F.3, negativen F.3, F.4
12	F.1, F.3, F.5
13	F.1, F.3, F.5, F.4
14	F.1, F.3, F.5, negativen F.3
15	F.1, F.3, F.5, negativen F.3, F.4
16	F.1, F.2
17	F.1, F.2, F.4
18	F.1, F.2, negativen F.3
19	F.1, F.2, negativen F.3, F.4
20	F.1, F.2, F.5

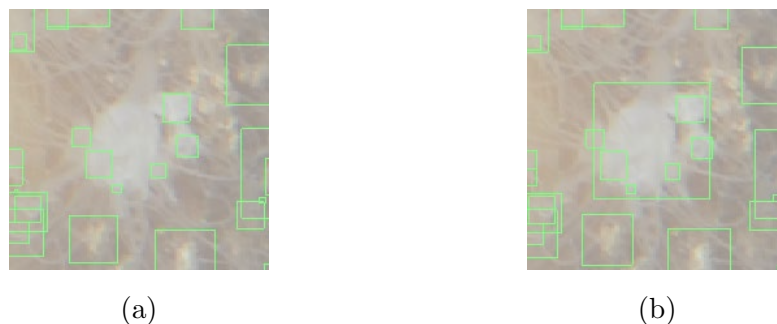
Tabela 4.2: Kombinacije metod.

Št. kombinacije	uporabljene metode
21	F.1, F.2, F.5, F.4
22	F.1, F.2, F.5, negativen F.3
23	F.1, F.2, F.5, negativen F.3, F.4
24	F.1, F.2, F.3
25	F.1, F.2, F.3, F.4
26	F.1, F.2, F.3, negativen F.3
27	F.1, F.2, F.3, negativen F.3, F.4
28	F.1, F.2, F.3, F.5
29	F.1, F.2, F.3, F.5, F.4
30	F.1, F.2, F.3, F.5, negativen F.3
31	F.1, F.2, F.3, F.5, negativen F.3, F.4

Tabela 4.3: Kombinacije metod.



Slika 4.10: Graf priklica v odvisnosti postavljene meje IOU za različne kombinacije filtrov.



Slika 4.11: a) Slika polipa pri izvedbi z manjšim parametrom M zaradi česar predlaganje regije ni bilo uspešno. b) Uporaba višje vrednosti parametra M s katero je detekcija polipa uspešna.

do velikosti M . Seveda je lahko radialnost neke regije, ki ima radij večji od M dovolj visoka tudi pri preverjanju radialnosti manjših velikosti vendar pa je nato končna predlagana regija manjša od dejanske velikosti iskane kar pa vpliva na končni rezultat. Lahko se zgodi tudi, da zaradi premajhnega M nekatere regije zgrešimo kot lahko vidimo na Sliki 4.11.

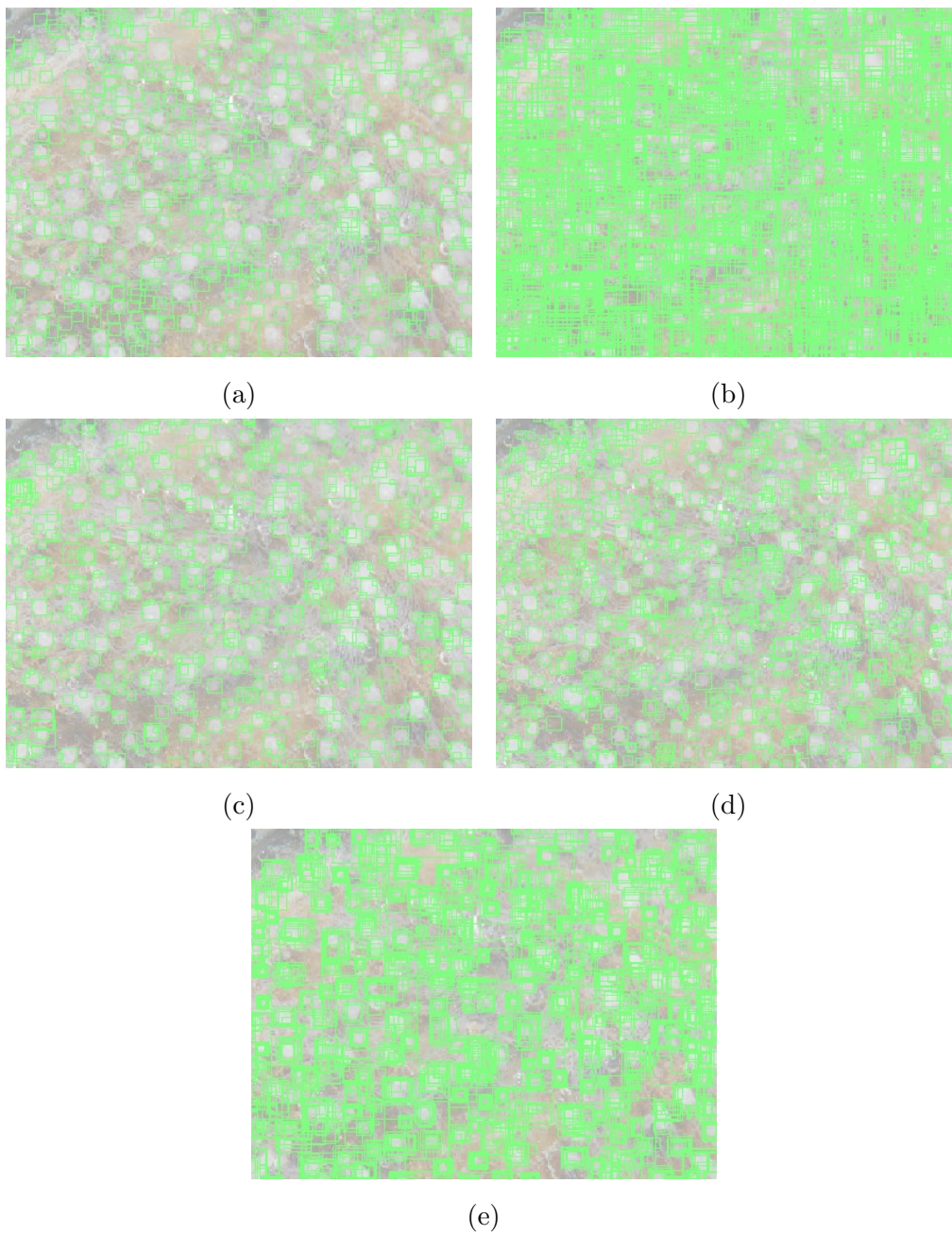
4.4 Primerjava rezultatov

Težko je oceniti, da je ena metoda pri reševanju problema boljša od druge, saj se uspeh pristopa lahko ocenjuje na veliko različnih načinov. Različni algoritmi predlagajo zelo različna števila regij. V primeru, da bi bila hitrost pomembna bi seveda dali prednost metodam, ki generirajo manjša števila regij. Če bi želeli maksimizirati povprečno preciznost pri klasifikaciji kompleksnejših objektov bi imele prednost metode, ki dosegaajo višjo stopnjo priklica pri visokih vrednostih IOU [22]. Slika 4.13 prikazuje povprečne rezultate obravnavanih metod na zbirki slik. Metodi RRD ter RRD brez uporabe morfoloških operacij dosejata zelo podobno uspešnost. Slednja predlaga veliko več potencialnih regij in tako dosega rahlo boljše rezultate sploh pri manjših vrednostih IOU. Pri stopnji $IOU = 0.3$ je priklic metod v istem vrstnem redu 0.9267 ter 0.9035. Pri stopnji $IOU = 0.2$ pa 0.9759 ter 0.9557. Ob nižjih

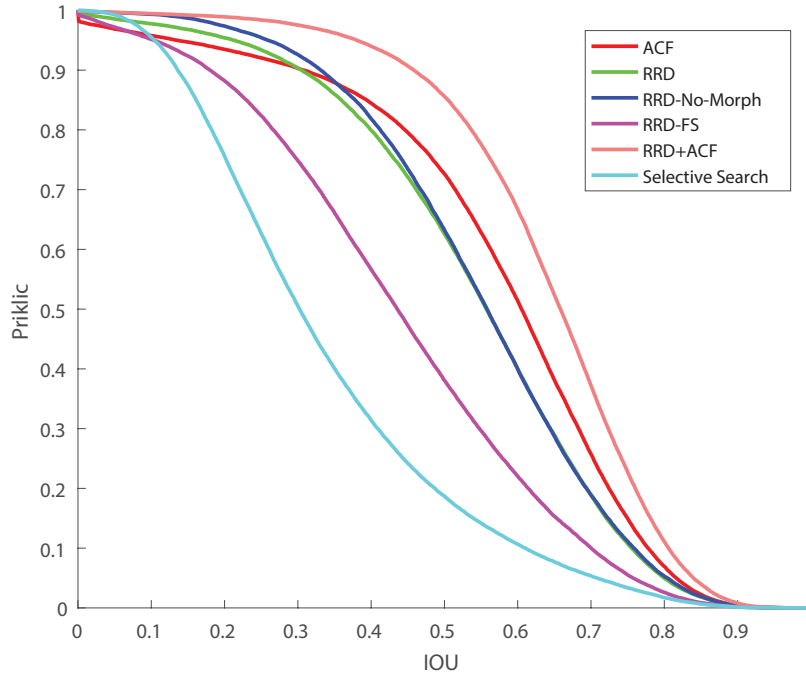
Metoda	Št. regij	AR3	$IOU > 0.5$	$IOU > 0.3$
RRD	133169	0.5028	0.6253	0.9040
RRD-NM	234779	0.5109	0.6330	0.9267
ACF	97662	0.5721	0.6936	0.9035
RRD-FS	231828	0.3313	0.4113	0.7503
SS	786148	0.1825	0.1850	0.5080
RRD+ACF	230831	0.6816	0.8574	0.9755

Tabela 4.4: Podatki uspešnosti metod.

mejah IOU metoda RRD deluje zelo dobro saj pokrije skoraj vse anotirane polipe vendar pa je njena lokalizacija slabša od tiste pri metodi ACF, ki pri zahtevnejših mejah IOU dosega veliko boljše rezultate. Najboljše rezultate dosega kombinacija metod ACF in RRD. V Tabeli 4.4 so zapisani podatki o uspešnosti obravnavanih metod. Za vsako metodo je zapisano število regij, ki jih je metoda generirala, vrednost AR3 na podatkovni zbirki, priklic pri meji $IOU \geq 0.5$ ter priklic pri meji $IOU \geq 0.3$. Na Sliki 4.12 so prikazane predlagane regije na delu slike za vsako obravnavano metodo.



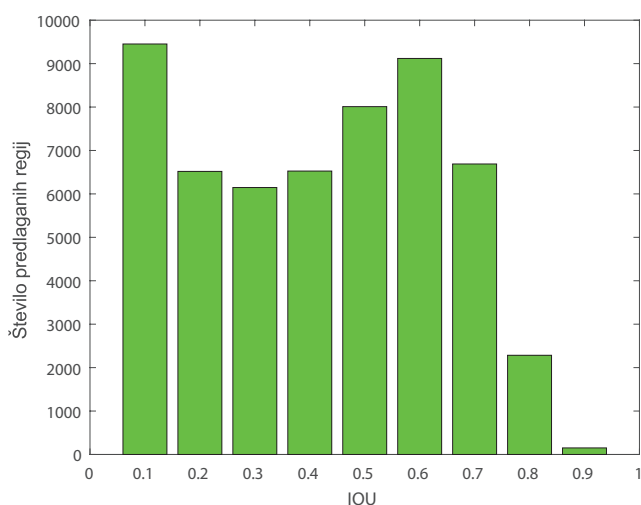
Slika 4.12: Predlagane regije na delu slike. a) Predlagane regije algoritma ACF. b) Predlagane regije metode Selective Search. c) Predlagane regije metode RRD. d) Predlagane regije metode RRD brez morfoloških operacij. e) Predlagane regije kombinacije metod RRD in Felzenszwalb.



Slika 4.13: Povprečna uspešnost obravnavanih algoritmov na isti podatkovni zbirki.

4.5 Rezultati metode ACF

Rezultate pridobljene z metodo ACF smo uporabili za primerjavo ostalih pristopov in nismo izvajali dodatnih eksperimentov za optimizacijo uspešnosti metode. Vedeli smo, da pristop deluje dobro iz dela [41]. Z vrednostjo AR3 0.5721 je po tej meri metoda boljša od metode RRD. Prednost algoritma ACF pred ostalimi je majhna količina slabših regij, ki jih predlaga. Predlaga 97662 potencialnih območij polipov torej le 2.611 na anotacijo vendar z le marginalno razliko v uspešnosti glede na RRD. Dosega tudi veliko količino regij z dobro pokritostjo saj je kar 69.36% anotacij prekritih z regijami z mero IOU višjo od 0.5.

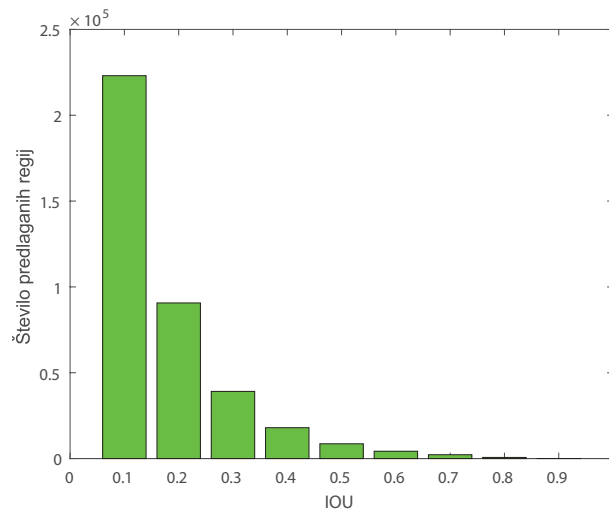


Slika 4.14: Histogram prikazuje število regij glede na IOU za metodo ACF.

4.6 Rezultati Selective Search

Selective Search je pri aplikaciji na dani problem predstavljal težave, saj je morala biti že začetna segmentacija dovolj majhna, da ni pokrivala večih polipov hkrati. To je metodo upočasnilo zahtevala pa je tudi velike količine spomina. Za pridobivanje rezultatov, ki so vsaj približno tako dobri kot pri ostalih metodah je bilo sliko potrebno obdelati večkrat z uporabo različnih parametrov, ki vplivajo na potek algoritma. Regije, ki so bile rezultat vsakega izvajanja algoritma smo potem združili in ocenjevali kot ostale metode. Zaradi združevanja tako predlaganih regij je algoritem za vsako sliko predlagal več sto tisoč lokacij. Kljub veliki količini regij, algoritem ni dosegal želenih rezultatov.

Čeprav algoritem dosega odlične rezultate na podatkovnih zbirkah PASCAL VOC2007, MS COCO 2014 in ImageNet 2013 [21], pri problemu detekcije polipov ni pokazal obetujočih rezultatov. Razlog za slabo uspešnost pri problemu bi lahko bil sam način delovanja Selective Search. Po začetni segmentaciji na manjše regije z uporabo [16], združuje te regije v večje vendar



Slika 4.15: Histogram prikazuje število regij glede na IOU za metodo Selective Search.

pa so polipi že sami po sebi zelo majhne regije zaradi česar združevanje v večje regije ne bi imelo smisla. Prav tako so meje med posameznimi polipi velikokrat zabrisane zaradi česar bi se posamezne regije po več merah podobnosti ujemale med seboj ter se tako hitro združevale kar pa je za štetje polipov nezaželeno. Algoritem na celotni podatkovni zbirki generira 786148 regij oziroma 21.02 predlogov za vsako anotacijo. Doseže $AR3$ vrednost 0.1825. Z dodatnim testiranjem in spreminjanjem parametrov bi verjetno lahko pridobili boljše rezultate. Prav tako bi lahko glede na znanje, ki ga imamo o velikosti polipov odstranili večino predlaganih regij.

4.7 Analiza metode RRD

Po nastavitvi parametrov smo z metodo dosegli dobre rezultate, vendar pa se uspešnost spreminja glede na sliko, ki jo uporabimo za generiranje regij. Eden izmed razlogov za to je napačna nastavitev parametrov za posamezno sliko. Poskušali smo nastaviti parametre, ki bi dobro delovali na vseh slikah vendar

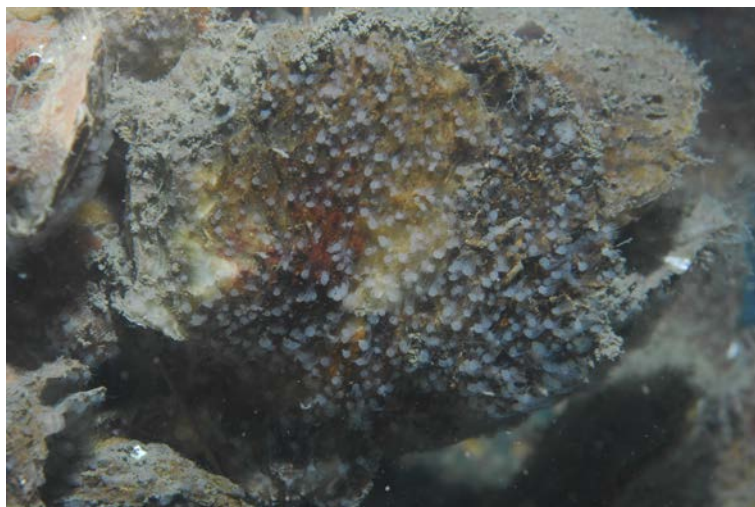


Slika 4.16: Vidimo, da je prekrivanje polipov na sliki problematično saj je iskati meje posameznih polipov težko že za človeka. Težavno je tudi veliko število in pa majha velikost polipov.

pa je to težavno zaradi številnih razlik med slikami. Povprečna velikost polipov se med slikami močno razlikuje. Kot primer uporabimo Sliki 4.17 in 4.16 kjer lahko vidimo kako zelo se med slikama razlikuje velikost polipov. Vidimo lahko tudi, da se polipi na Sliki 4.16 veliko bolj prekrivajo. Kljub temu so rezultati dobri in je po uspešnosti primerljiva ali celo boljša od ostalih preverjanih pristopov.

4.7.1 Predlagane regije in povprečna uspešnost

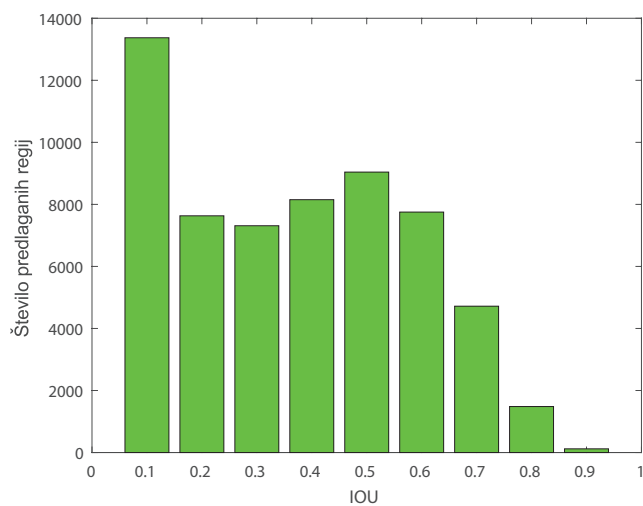
Metoda RRD je na podatkovni bazi slik predlagala 133169 regij polipov oziroma 3.56 regij na označbo polipa. Povprečna mera AR3 je dosegla vrednost 0.5028. Obravnavali pa smo tudi verzijo metode RRD, ki maske ne obdela z morfološki operacijami odpiranja in zapiranja. Zaradi tega vsebuje maska veliko več šuma in tako predlaga skoraj dvakrat višje število regij kot pa verzija metode, ki morfološki operaciji uporablja. Predlaga 234779 regij oziroma 6.28 regij na anotacijo vendar pa dosega boljše rezultate. Povprečna



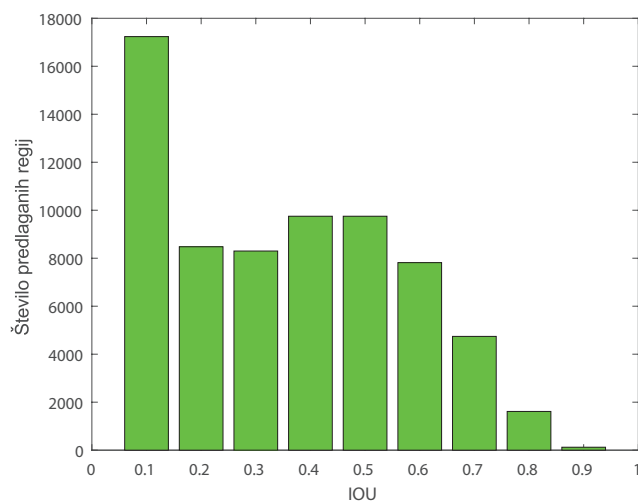
Slika 4.17: Polipi na sliki so relativno veliki, večinoma dobro ločeni eden od drugega in so na podlagi, ki se po barvi večinoma razlikuje od njih samih. Takšni pogoji bistveno olajšajo detekcijo.

vrednost AR3 je 0.5109.

Število predlaganih regij glede na kriterij prekrivanja IOU na celotni podatkovni bazi je prikazano v Slikah 4.18 in 4.19. Histograma ne vsebujeta regij z vrednostjo IOU manj kot 0.1 zaradi ogromne količine regij, ki spadajo v to kategorijo kar bi poslabšalo preglednost histogramov. Vsak stolpec histograma predstavlja število predlaganih regij katerih vrednost IOU je višja od števila napisanega pod stolpcem in nižja od števila napisanega pod stolpcem na desni.



Slika 4.18: Histogram prikazuje število regij glede na IOU za metodo RRD.

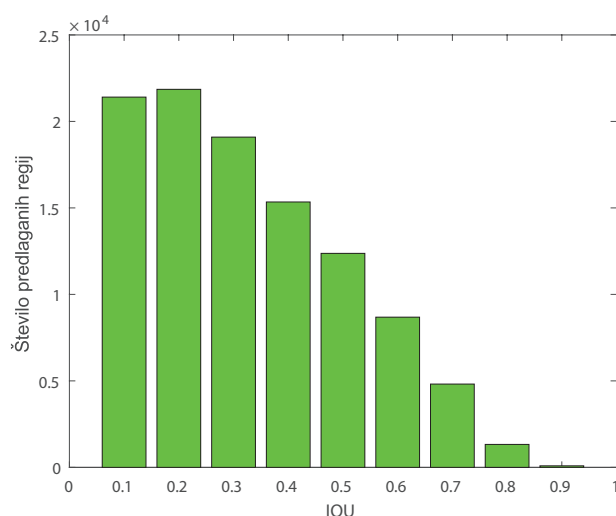


Slika 4.19: Histogram prikazuje število regij glede na IOU za metodo RRD brez morfoloških operacij.

4.8 Rezultati kombinacije metod RRD in Felzenszwalb

Metodo RRD smo kombinirali z algoritmom Felzenszwalb, da bi maskirano sliko, prikazano na Sliki 3.7, uspeli še dodatno segmentirati glede na robove med polipi, ki jih maska pokriva skupaj. Odvisno od nastavljenih parametrov algoritma Felzenszwalb ter generirane maske je algoritem sliko segmentiral preveč ali pa premalo. Boljše rezultate smo seveda dosegli s parametri, pri katerih algoritem predlaga večjo količino regij. Algoritem je v tem primeru na podatkovni bazi predlagal 231828 regij oziroma 6.19 na anotacijo. Povprečna vrednost AR3 pa je dosegla 0.3313. Z bolj natančnim iskanjem parametrov bi verjetno lahko rezultate do neke mere izboljšali.

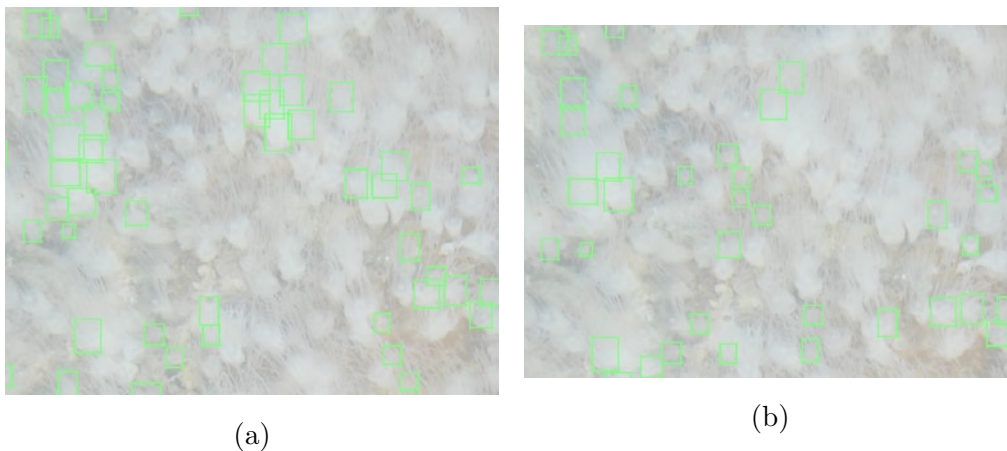
Rezultati so močno odvisni od izbranih parametrov za generiranje maske ter segmentacije. Izbira parametrov za generiranje maske ter pravih morfoloških operacij je v tem primeru zelo pomembna saj z masko, ki vsebuje veliko šuma, sprejemljivih rezultatov, zaradi slabe predstavitve delov slike, ki predstavljajo polipe, ni mogoče dobiti. Za dobre rezultate je potrebna maska podobna Sliki 4.9, kjer so regije polipov dobro pokrite, segmentacija med njimi pa ni tako pomembna, saj jo kasneje izvaja drugi del algoritma.



Slika 4.20: Histogram prikazuje število regij glede na IOU za kombinacijo metod RRD in Felzenszwalb.

4.9 Kombinacija metod ACF in RRD

Združili smo predloge metod ACF in RRD ter jih ocenili, kot da bi predloge generirala ene sama metoda. S kombinacijo obeh metod smo dosegli zelo dobre rezultate. Z vrednostjo AR3 0.6816 dosega kombinirana metoda najboljši rezultat med obravnavanimi pristopi. Metoda generira 230831 predlogov, predlaga pa tudi visoko število regij z zelo dobrim pokritjem, pri vrednosti IOU nad 0.5 je priklic 0.8574. Tudi pri nižjih vrednostih IOU dosega metoda dobre rezultate saj pri vrednosti IOU nad 0.3 dosega priklic 0.9755. Metodi ACF in RRD delujeta komplementarno saj velikokrat predlagata različne regije in tako zmanjšata število anotacij, ki jih ne pokrije nobena od metod. Kot vidimo na Sliki 4.21, se veliko slabo pokritih anotacij razlikuje med metodama zaradi česar se ob združitvi predlaganih regij obeh metod rezultat izboljša.



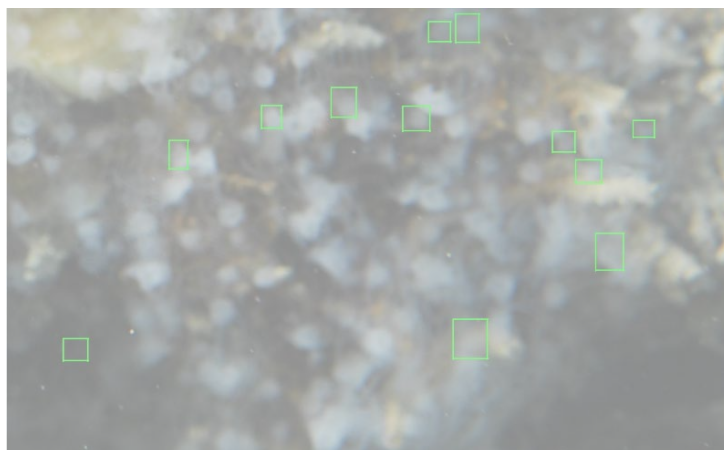
Slika 4.21: Slabo pokrite anotacije za metodi ACF in RRD. a) Anotacije, ki jih metoda ACF pokrije z manjšo vrednostjo IOU. b) Anotacije, ki jih metoda RRD pokrije z manjšo vrednostjo IOU.

4.10 Prednosti in slabosti metode RRD

Metoda RRD je na naši podatkovni bazi dosegla dobre rezultate sploh glede na težavnost naloge zaradi velikih razlik med slikami, majhnosti polipov, njihovega prekrivanja, zlivanja z okoljem in zamegljenosti. Težave algoritmu povzročajo predvsem polipi, ki so za detekcijo problematični tudi za ljudi. Velikokrat slabo predlaga lokacijo polipov, ki se med seboj prekrivajo ali pa se jih težko loči od ozadja zaradi njihove prosojnosti ali pa zamegljenosti. Nekaj napake vseh preizkušenih pristopov bi lahko pripisali tudi nekaterim vprašljivim anotacijam, ki bi lahko nastale zaradi človeške napake. Na Slikah 4.22 in 4.23 so prikazani deli slike s problematičnimi anotacijami na katerih nobena predlagana regija ne preseže $IOU = 0.2$. Na Sliki 4.22 lahko vidimo, da je polipe izjemno težko ločevati že za ljudi saj se zaradi enotne barve in slabše osvetlitve zlijejo v večjo homogeno regijo. Na Sliki 4.23 pa je očitno, da je ob takšni zamegljenosti slike, zaradi napak fotografa, težko z veliko gotovostjo določiti obstoj polipa. Takih anotacij je malo in dobra lastnost metode RRD je, da ima pri nižjih stopnjah IOU zelo visok priklic.



Slika 4.22: Z zelenimi pravokotniki so označene anotacije, ki so za natančno detekcijo težavne zaradi prekrivanja polipov.



Slika 4.23: Z zelenimi pravokotniki so označene anotacije, ki so za natančno detekcijo težavne zaradi zamegljenosti polipov.

Poglavje 5

Sklep

Naslovili smo problem detekcije polipov na slikah ostrig. Osredotočili smo se predvsem na predlaganje hipotetičnih regij za kasnejšo uporabo z močnim klasifikatorjem. Predlagali smo metodo RRD, ki z uporabo usmerljivih filtrov [17] ter mere radialnosti [31] predlaga potencialne regije, kjer naj bi se polipi na sliki nahajali. Izvedli smo tudi primerjavo uspešnosti algoritmov RRD, Selective Search [46], ACF [11] ter kombinacije algoritma RRD ter Felzenszwalb [16] pri reševanju problema predlaganja regij za detekcijo polipov na slikah ostrig ter izpostavili prednosti in slabosti posameznih pristopov. Preizkusili smo možnost aplikacije algoritma RRD na dani problem ter poskušali optimizirati metodo za čimboljše delovanje. Predlagali smo tudi mero za ocenjevanje uspešnosti posamezne metode pri reševanju problema detekcije polipov AR3, ki uspešnost meri z deležom pokritih ročnih anotacij pri vrednosti $IOU > 0.3$.

Izvedli smo analizo rezultatov RRD, RRD brez morfoloških operacij, kombinacije RRD in segmentacije Felzenszwalb [16], Selective Search [46] ter ACF [11]. Ocenili smo tudi uspešnost kombinacije metod RRD in ACF. Metode smo izvedli na isti podatkovni zbirki slik polipov ter nato izračunali njihovo uspešnost. Z uporabo metod Selective Search in kombinacije RRD in Felzenszwalb nam ni uspelo dobiti rezultatov, ki bi se po uspešnosti pri reševanju

danega problema lahko primerjali z ostalimi. Potrdili smo ugotovitev [41], da je algoritem ACF zelo uspešen pri predlaganju hipotetičnih regij polipov, saj po naši meri uspešnosti AR3 dosega najvišjo vrednost od vseh obravnavanih metod. Metodi RRD in RRD brez morfoloških operacij dosegata dobre rezultate glede na mero AR3 ter po priklicu anotacij z mejno vrednostjo $IOU = 0.3$ prekašata ACF.

Na uspešnost metode RRD najbolj vplivajo parametri, ki vplivajo na generiranje maske v prvem delu algoritma. Izbor optimalnih parametrov je težaven zaradi medsebojne odvisnosti spremenljivih faktorjev, ki vplivajo na rezultat algoritma zato seveda obstaja možnost, da bi se rezultat metode RRD še izboljšal z nastavitvijo boljše kombinacije parametrov.

5.1 Možne izboljšave

Zanimivo bi bilo videti rezultate naučenega klasifikatorja, na primer takega, kot se uporablja v delu [41], z uporabo regij, ki jih predlaga RRD. Glede na veliko količino predlaganih regij, bi bilo zanimivo videti čas, ki bi ga za obdelavo slik potreboval, saj bi tako imeli več informacij o pomembnosti števila predlaganih regij, kar bi lahko vključili v mero za ocenjevanje posameznih pristopov. Obdelava regij s klasifikatorjem bi nam podala tudi informacijo o pomembnosti stopnje prekrivanja za uspešno klasifikacijo pri danem problemu. Tudi ta podatek bi lahko uporabili pri prilagajanju metode za boljše delovanje, omogočil pa bi nam tudi baziranje meje IOU za ocenjevanje uspešnosti pristopov na eksperimentalnih podatkih. Sedaj smo namreč meje IOU pri meri AR3 osnovali na podlagi vizualne analize. Metoda RRD bi lahko bila izboljšana tudi z boljšimi parametri, saj je zaradi velikega števila možnosti nastavitve parametrov malo verjetno, da so bili pri eksperimentih nastavljeni optimalni parametri. Isto velja tudi za pristope Selective Search [46] ter RRD v kombinaciji z algoritmom Felzenszwalb [16]. Kljub možnim izboljšavam smo dosegli dobre rezultate na problemu, ki je, kot veliko drugih problemov detekcije objektov, težko rešljiv.

Literatura

- [1] A. Ramšak L. Catenacci A. Malej, T. Kogovšek. Blooms and population dynamics of moon jellyfish in the northern adriatic. *Cahiers de Biologie Marine*, 53(3):337–342, 2011.
- [2] Robiyanti Adollah, MY Mashor, NF Mohd Nasir, H Rosline, H Mahsin, and H Adilah. Blood cell image segmentation: a review. In *4th Kuala Lumpur International Conference on Biomedical Engineering 2008*, pages 141–144. Springer, 2008.
- [3] Yousef Al-Kofahi, Wiem Lassoued, William Lee, and Badrinath Roodsam. Improved automatic detection and segmentation of cell nuclei in histopathology images. *IEEE Transactions on Biomedical Engineering*, 57(4):841–852, 2010.
- [4] Carlos Arteta, Victor Lempitsky, J Alison Noble, and Andrew Zisserman. Learning to detect cells using non-overlapping extremal regions. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 348–356. Springer, 2012.
- [5] Lucas Assirati, Núbia Rosa da Silva, Lilian Berton, Alneu de A Lopes, and Odemir M Bruno. Performing edge detection by difference of gaussians using q-gaussian kernels. In *Journal of Physics: Conference Series*, volume 490, page 12020. IOP Publishing, 2014.

-
- [6] Boštjan Bohte. Sistem za delno avtomatsko štetje polipov na slikah. B.s. thesis, Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, Februar 2015.
 - [7] Felix Buggenthin, Carsten Marr, Michael Schwarzfischer, Philipp S Hoppe, Oliver Hilsenbeck, Timm Schroeder, and Fabian J Theis. An automatic method for robust and fast cell detection in bright field images from high-throughput microscopy. *BMC bioinformatics*, 14(1):297, 2013.
 - [8] John Canny. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6):679–698, 1986.
 - [9] Ming-Ming Cheng, Ziming Zhang, Wen-Yan Lin, and Philip H. S. Torr. BING: Binarized normed gradients for objectness estimation at 300fps. In *IEEE CVPR*, 2014.
 - [10] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893. IEEE, 2005.
 - [11] Piotr Dollár, Ron Appel, Serge J. Belongie, and Pietro Perona. Fast feature pyramids for object detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(8):1532–1545, 2014.
 - [12] Piotr Dollár, Serge Belongie, and Pietro Perona. The fastest pedestrian detector in the west. 2(3):7, 2010.
 - [13] Piotr Dollar, Zhuowen Tu, Pietro Perona, and Serge Belongie. Integral channel features. In *Proc. BMVC*, pages 91.1–91.11, 2009. doi:10.5244/C.23.91.
 - [14] Piotr Dollár and C. Lawrence Zitnick. Fast edge detection using structured forests. *ArXiv*, 2014.

- [15] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–1645, 2010.
- [16] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient graph-based image segmentation. *IJCV*, (59), 2004.
- [17] William T. Freeman and Edward H. Adelson. The design and use of steerable filters. *IEEE transactions on pattern analysis and machine intelligence*, 13(9), 1991.
- [18] Yoav Freund and Robert E Schapire. A desicion-theoretic generalization of on-line learning and an application to boosting. In *European conference on computational learning theory*, pages 23–37. Springer, 1995.
- [19] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1440–1448, 2015.
- [20] Ji Wan Han, Toby P Breckon, David A Randell, and Gabriel Landini. The application of support vector machine classification to detect cell nuclei for automated microscopy. *Machine Vision and Applications*, 23(1):15–24, 2012.
- [21] Jan Hendrik Hosang, Rodrigo Benenson, Piotr Dollár, and Bernt Schiele. What makes for effective detection proposals? *CoRR*, abs/1502.05082, 2015.
- [22] Jan Hendrik Hosang, Rodrigo Benenson, and Bernt Schiele. How good are detection proposals, really? *CoRR*, abs/1406.6962, 2014.
- [23] Mathews Jacob and Michael Unser. Optimal steerable filters for feature detection. In *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, volume 3, pages III–433. IEEE, 2003.
- [24] David Jacobs. Image gradients, class notes for cmsc 426, 2005.

-
- [25] Joseph B Kruskal. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American Mathematical society*, 7(1):48–50, 1956.
 - [26] T. Lindeberg. Scale invariant feature transform. *Scholarpedia*, 7(5):10491, 2012.
 - [27] Tony Lindeberg. Image matching using generalized scale-space interest points. *Journal of Mathematical Imaging and Vision*, 52(1):3–36, 2015.
 - [28] David G Lowe. Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, pages 1150–1157. Ieee, 1999.
 - [29] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.
 - [30] Santiago Manen, Matthieu Guillaumin, and Luc Van Gool. Prime object proposals with randomized prim’s algorithm. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2536–2543, 2013.
 - [31] Jasna Maver. Self-similarity and points of interest. *IEEE transactions on pattern analysis and machine intelligence*, 32(7), 2010.
 - [32] Tim McInerney and Demetri Terzopoulos. Deformable models in medical image analysis: a survey. *Medical image analysis*, 1(2):91–108, 1996.
 - [33] Erik Meijering. Cell segmentation: 50 years down the road. *IEEE Signal Processing Magazine*, 29(5), 2012.
 - [34] Constantine Papageorgiou and Tomaso Poggio. A trainable system for object detection. *International Journal of Computer Vision*, 38(1):15–33, 2000.

-
- [35] GPM Priyankara, OW Seneviratne, RKOH Silva, WVD Soysa, and CR De Silva. An extensible computer vision application for blood cell recognition and analysis.
 - [36] J. Ross Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, 1986.
 - [37] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.
 - [38] Jos BTM Roerdink and Arnold Meijster. The watershed transform: Definitions, algorithms and parallelization strategies. *Fundamenta informaticae*, 41(1, 2):187–228, 2000.
 - [39] Robert E Schapire. The strength of weak learnability. *Machine learning*, 5(2):197–227, 1990.
 - [40] Robert E Schapire. Explaining adaboost. In *Empirical inference*, pages 37–52. Springer, 2013.
 - [41] Kristijan Shirgoski. Computer-vision based polyp detection in under-water images. B.s. thesis, University of Ljubljana, Faculty of computer science, September 2015.
 - [42] Eero P Simoncelli and Hany Farid. Steerable wedge filters. In *Computer Vision, 1995. Proceedings., Fifth International Conference on*, pages 189–194. IEEE, 1995.
 - [43] Schuyler Smith. *CS231B: The Cutting Edge of Computer Vision*. Stanford University, June 2015.
 - [44] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders. Selective search for object recognition. *International Journal of Computer Vision*, 104(2):154.

- [45] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders. Selective search for object recognition. *International Journal of Computer Vision*, 104(2):154–171, 2013.
- [46] K. E. A. van de Sande, J. R. R. Uijlings, T. Gevers, and A. W. M. Smeulders. Segmentation as selective search for object recognition. In *IEEE International Conference on Computer Vision*, 2011.
- [47] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–511. IEEE, 2001.
- [48] Paul Viola and Michael J Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.
- [49] C. Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*, 2014.